8-1974

Analysis of Unsteady Flow Toward Artesian Wells by Three-Dimensional Finite Elements

Digital Object Identifier: https://doi.org/10.13023/kwrri.rr.75

Yang H. Huang
University of Kentucky

Shen-Jyh Wu
University of Kentucky

Research Report No. 75

ANALYSIS OF UNSTEADY FLOW TOWARD ARTESIAN WELLS BY THREE-DIMENSIONAL FINITE ELEMENTS

Dr. Yang H. Huang
Principal Investigator

Graduate Student Assistant: Shen-Jyh Wu

Project Number A-042-KY (Completion Report)
Agreement Numbers 14-31-0001-3517 (FY 1972)
14-31-0001-3817 (FY 1973)
14-31-0001-4017 (FY 1974)
Period of Project - November, 1971 - June, 1974

University of Kentucky Water Resources Institute
Lexington, Kentucky

The work on which this report is based was supported in part by funds provided by the Office of Water Resources Research, United States Department of the Interior, as authorized under the Water Resources Research Act of 1964.

August, 1974

ABSTRACT

A three-dimensional finite element computer program was developed for analyzing unsteady flow toward artesian wells. The program is designed especially for determining the drawdown around an artesian well penetrating fully or partially a nonhomogeneous and anisotropic aquifer of irregular shape and cross section. It can also be used as a general program for aquifer simulation and evaluation. A major advantage of the program lies in the minimum amount of input data required. By assuming the top and bottom boundaries of the aquifer as two arbitrary planes, the aquifer will be divided into six- or eight-node elements, and their nodal coordinates generated automatically. The program was well documented and can be used for solving complex problems encountered in practice.

The results of this study indicate that unsteady flow toward artesian wells can be analyzed effectively by three-dimensional finite elements. A comparison between the finite element and the exact mathematical solutions for a simple case shows that both solutions check closely. The solution obtained from the computer program for a complex case involving a nonhomogeneous aquifer was checked against that obtained previously by the use of cylindrical elements, and both are found in good agreements. The program was applied to a variety of cases, and reasonable results were obtained.

KEYWORDS: anisotropy; artesian well*; computer program*; computer simulation; confined aquifer; digital computer; drawdown*; finite element analysis*; groundwater; heterogeneity; numerical analysis; unsteady flow*

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Page

CHAPTER I

INTRODUCTION

As stated in the original proposal, the objective of this research was to develop a three-dimensional finite element method for analyzing unsteady flow toward artesian wells, and the end product would be a computer program available to field engineers and scientists for determining the drawdown around an artesian well. It was further stipulated that the program should be applicable to anisotropic and nonhomogeneous aquifers of irregular shape and cross section, and that by simply reading into a computer the dimensions of the aquifer and the well, the various formation constants, and the boundary conditions, the drawdown around the well at any given time could be obtained. As will be described in this report, this objective has been accomplished.

The application of the finite element method for analyzing unsteady flow toward an artesian well has been a subject of considerable studies in recent years [Janvendel and Witherspoon, 1968, 1969; Pinder and Frind, 1972; Huang, 1972, 1973]. Because of the large amount of computer time and storage required, most of these applications have so far been limited to two-dimensional flow only. Very little work has been done in analyzing flow in three dimensions. The theory of three-dimensional finite element method is well known. The problem now on hand is how to incorporate the method into a computer program which can be used to model realistically a three-dimensional aquifer.

As a preliminary step toward the above goal, a simplified three-dimensional computer program was developed using cylindrical elements. Although the use of cylindrical elements greatly facilitates the formation of stiffness matrix and the generation of nodal coordinates, it has the limitation that accurate results can be obtained only when the actual hydrogeologic boundaries can be matched by the cylindrical elements. Details of the cylindrical elements was presented elsewhere [Sonnenfeld, 1973; Huang and Sonnenfeld, 1974] and will not be covered in this report. It is believed that the computer program presented in this report is more versatile and can match the aquifer boundaries more closely.

CHAPTER II

RESEARCH PROCEDURES

The general procedures for analyzing unsteady flow by finite elements are well known and have been illustrated in several textbooks [Zienkiewicz and Cheung, 1967; Remson et al., 1971; Desai and Abel, 1972]. Only those procedures directly related to the development of the computer program will be described here.

Governing Equations

The governing differential equation for unsteady flow can be written as

$$k_x \frac{\partial^2 h}{\partial x^2} + k_y \frac{\partial^2 h}{\partial y^2} + k_z \frac{\partial^2 h}{\partial z^2} - q = S_s \frac{\partial h}{\partial t} \tag{1}$$

in which x, y and z = cartesian coordinates; k_x, k_y and k_z = permeabilities in x, y and z directions, respectively; S_s = specific storage of the aquifer, which is defined as the amount of water in storage released from a unit volume of aquifer under a unit decline of head; q = time rate of discharge per unit volume, positive for discharge and negative for recharge; h = piezometric head; and t = time. If h_o is the initial piezometric head, the drawdown, s, can be defined as

$$s = h_o - h \tag{2a}$$

$$\text{or} \qquad h = h_o - s \tag{2b}$$

When the initial peizometric surface is a plane, as is assumed in this study, substitution of Eq. 2b into Eq. 1 results in a governing differential equation in terms of drawdown.

$$k_x \frac{\partial^2 s}{\partial x^2} + k_y \frac{\partial^2 s}{\partial y^2} + k_z \frac{\partial^2 s}{\partial z^2} + q = S_s \frac{\partial s}{\partial t} \tag{3}$$

To solve Eq. 3, the following initial and boundary conditions are specified: (1) the water is pumped out from the well at a constant rate; (2) the initial drawdown is everywhere zero before pumping started; and (3) part of the aquifer boundary may be impervious, part may be pervious with a constant discharge or recharge, and the other may maintain a zero drawdown.

Element Stiffness Matrix

In the finite element method, the aquifer is divided into a large number of elements, and only the drawdown at a discrete number of nodes is considered. To achieve a reasonable physical approximation of the actual aquifer, two types of finite elements are employed; one a six-node triangular prism, and the other a rectangular prism or a hexahedron, as shown in Figure 1. Both types of elements are formed by basic tetrahedra. The stiffness matrix of each tetrahedron is determined first and then superimposed to form the stiffness matrix of the composite element. The triangular prism is composed of three tetrahedra, while the hexahedron is composed of five. Because each composite element can be divided into two different ways, as shown in Figures 2 and 3, both divisions are used and the results averaged to obtain the stiffness matrix of the composite element. The use of average values greatly improves the accuracy of the results. The relationship between the nodal number of tetrahedron and the nodal number of triangular prism is presented in Table 1. The relationship for the hexahedron is presented in Table 2.

Table 1. Nodal relationship between tetrahedron and triangular prism.

Tetrahedron Number	Nodal number of tetrahedron			
	1	2	3	4
	Nodal number of triangular prism			
1	1	2	3	5
2	1	2	4	6
3	1	3	4	6
4	1	3	5	6
5	2	3	4	5
6	2	4	5	6

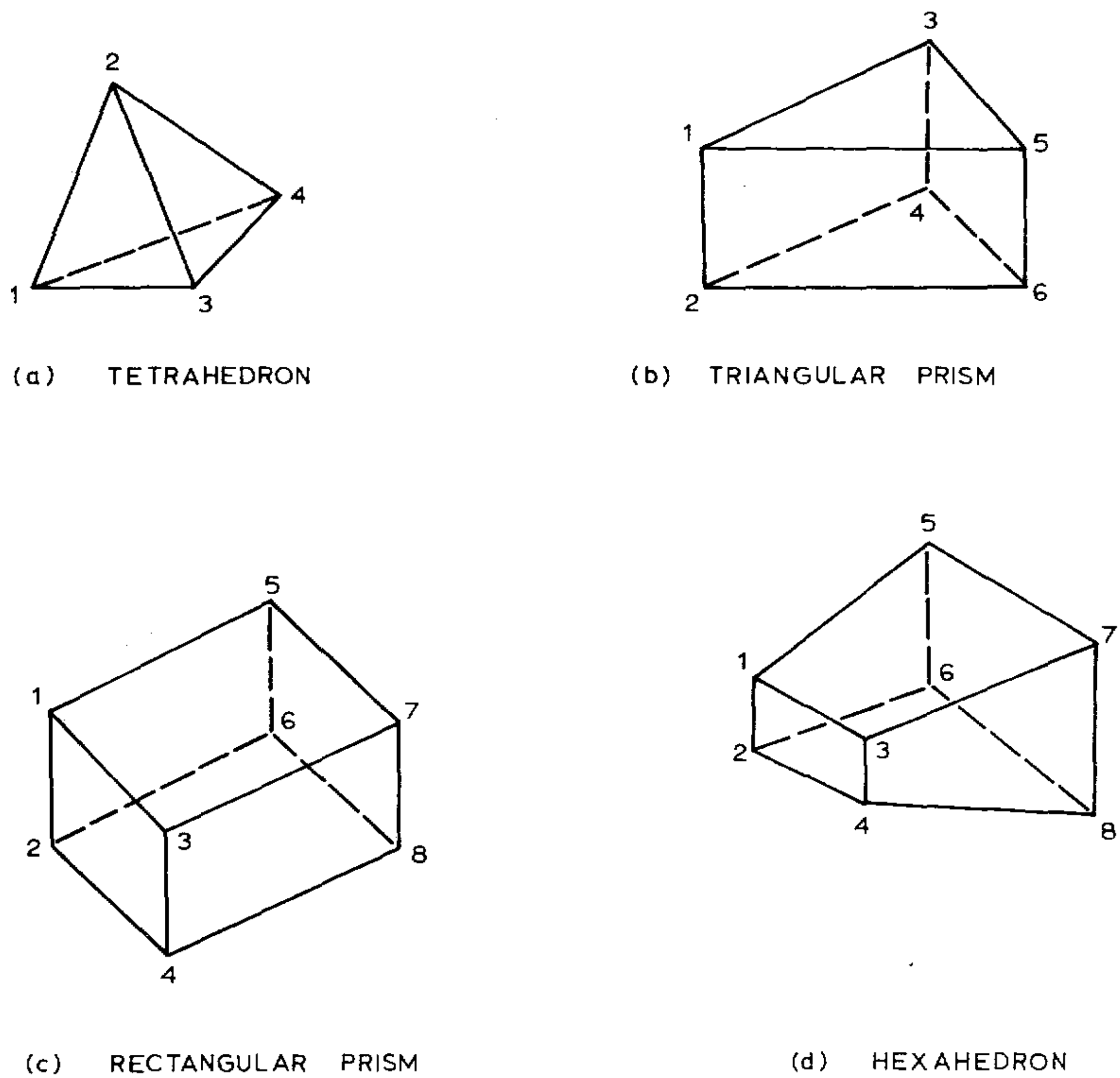


Figure 1. Types of three-dimensional elements and their numbering.

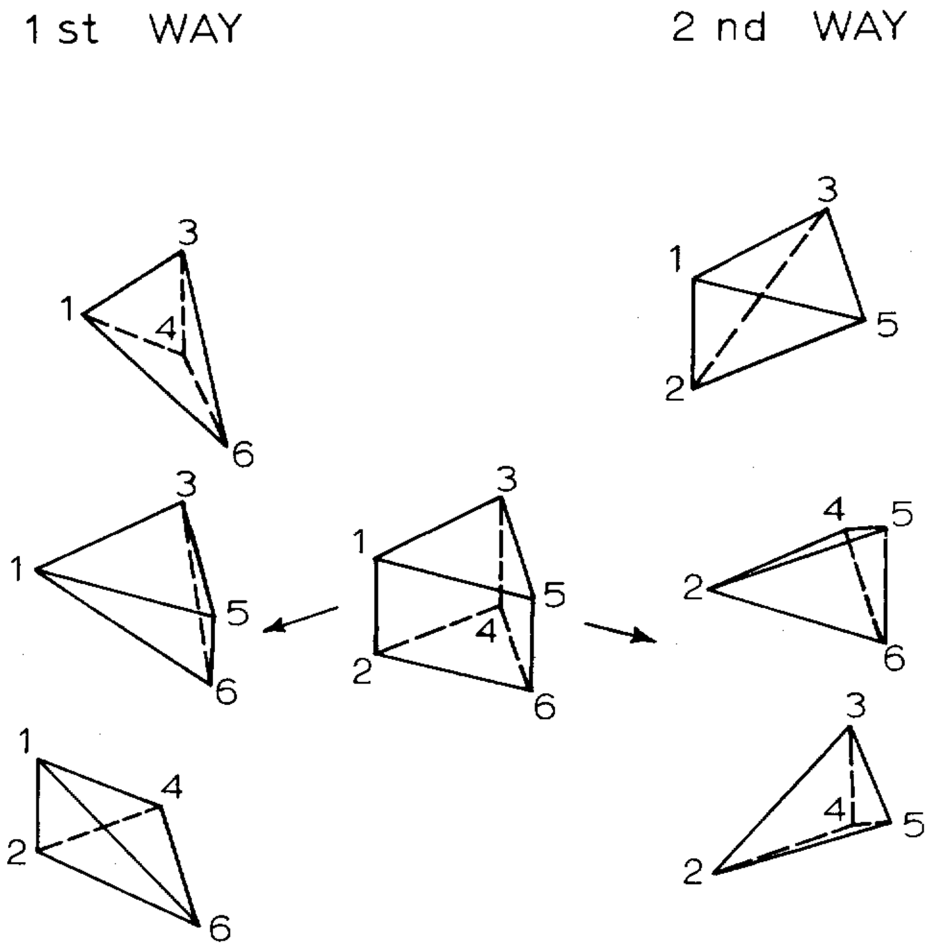


Figure 2. Subdivision of triangular prism into three tetrahedra.

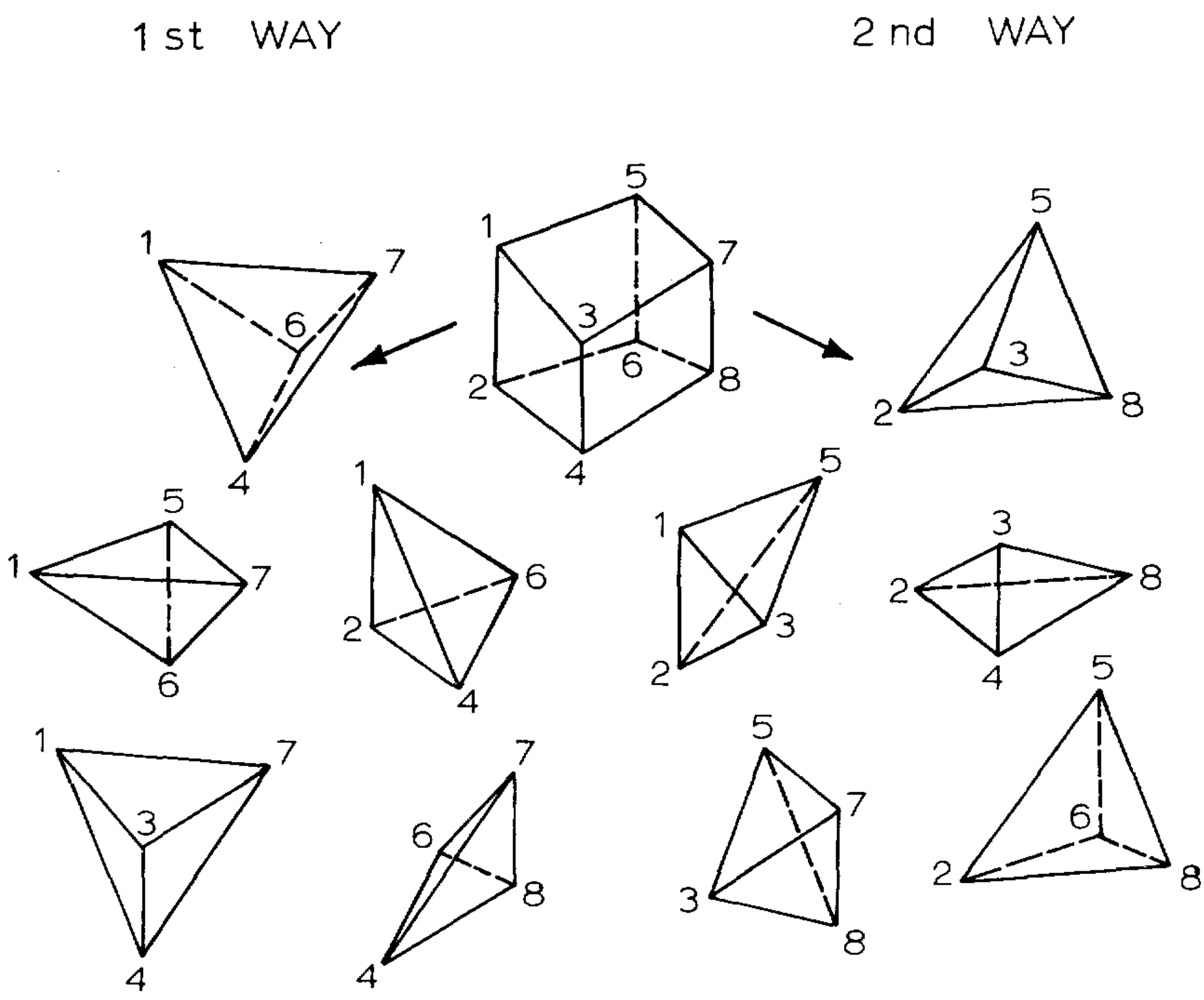


Figure 3. Subdivision of hexahedron into five tetrahedra.

Table 2. Nodal relationship between tetrahedron and hexahedron.

Tetrahedron Number	Nodal number of tetrahedron 1	2	3	4
	Nodal number of hexahedron			
1	1	3	4	7
2	1	5	6	7
3	1	4	6	7
4	1	2	4	6
5	4	6	7	8
6	3	5	7	8
7	2	3	5	8
8	1	2	3	5
9	2	5	6	8
10	2	3	4	8

According to the variational principle, the solution of Eq. 3 is equivalent to the minimization of the following integral equation with respect to s.

$$I = \int\int\int \left\{ 1/2 \ [k_x \left(\frac{\partial s}{\partial x}\right)^2 + k_y \left(\frac{\partial s}{\partial y}\right)^2 + k_z \left(\frac{\partial s}{\partial z}\right)^2 + S_s s \frac{\partial s}{\partial t} - qs] \right\} dxdydz \quad (4)$$

In Eq. 4, $\partial s / \partial t$ is an invariant when conditions for a particular instant are considered.

Assuming that the drawdown within a tetrahedron is a linear function of the x, y and z coordinates, the drawdown at any point can be expressed in terms of that at the nodes by

$$h = [N_1, \ N_2, \ N_3, \ N_4] \begin{Bmatrix} s_1 \\ s_2 \\ s_3 \\ s_4 \end{Bmatrix} \quad (5)$$

in which $N_i = (a_i + b_i x + c_i y + d_i z) / 6V$; s_i = drawdown at node i; i = a subscript indicating the ith node, which ranges from 1 to 4 as shown in Figure 1 a ; V = volume of tetrahedron which can be determined from the nodal coordinates by

$$V = \frac{1}{6}\begin{vmatrix} 1 & 1 & 1 & 1 \\ x_1 & x_2 & x_3 & x_4 \\ y_1 & y_2 & y_3 & y_4 \\ z_1 & z_2 & z_3 & z_4 \end{vmatrix} \tag{6}$$

and a_i, b_i, c_i and d_i = constants which can be determined from the nodal coordinates by

$$a_1 = (-1)^2 \det \begin{vmatrix} x_2 & x_3 & x_4 \\ y_2 & y_3 & y_4 \\ z_2 & z_3 & z_4 \end{vmatrix} \; ; \qquad b_1 = (-1)^1 \det \begin{vmatrix} 1 & 1 & 1 \\ y_2 & y_3 & y_4 \\ z_2 & z_3 & z_4 \end{vmatrix}$$

$$c_1 = (-1)^2 \det \begin{vmatrix} 1 & 1 & 1 \\ x_2 & x_3 & x_4 \\ z_2 & z_3 & z_4 \end{vmatrix} \; ; \qquad d_1 = (-1)^1 \det \begin{vmatrix} 1 & 1 & 1 \\ x_2 & x_3 & x_4 \\ y_2 & y_3 & y_4 \end{vmatrix} \tag{7}$$

The constants a_2, b_2, c_2 and d_2 can be obtained by replacing the subscripts or the exponents 1, 2, 3 and 4, as shown in Eq. 7, by 2, 3, 4 and 1 respectively. The same order or rotation is applied to obtain constants a_3, a_4, etc.

Minimizing Eq. 4 with respect to s_i gives

$$\frac{\partial I}{\partial s_i} = \int\int\int [k_x \frac{\partial s}{\partial x}\frac{\partial}{\partial s_i}\left(\frac{\partial s}{\partial x}\right) + k_y \frac{\partial s}{\partial y}\frac{\partial}{\partial s_i}\left(\frac{\partial s}{\partial y}\right) + k_z \frac{\partial s}{\partial z}\frac{\partial}{\partial s_i}\left(\frac{\partial s}{\partial z}\right)$$
$$+ S_s \frac{\partial s}{\partial t}\frac{\partial s}{\partial s_i} - q\frac{\partial s}{\partial s_i}]\, dxdydz = 0 \tag{8}$$

For a tetrahedral element with four nodes, four equations can be obtained from Eq. 8 to determine the four unknown nodal drawdowns. When Eq. 5 is substituted into Eq. 8, three matrices are obtained, i.e., a square matrix $[H]^e$ depending on permeability, a square matrix $[P]^e$ depending on specific storage, and a column matrix $\{F\}^e$ depending on discharge. The $[H]^e$ and $[P]^e$ are called the element stiffness matrix with the superscript e representing element. The coefficients of these matrices can be determined by

$$h_{ij} = \iiint \left[k_x \left(\frac{\partial N_i}{\partial x}\right)\left(\frac{\partial N_j}{\partial x}\right) + k_y \left(\frac{\partial N_i}{\partial y}\right)\left(\frac{\partial N_j}{\partial y}\right) + k_z \left(\frac{\partial N_i}{\partial z}\right)\left(\frac{\partial N_j}{\partial z}\right)\right] dxdydz \tag{9a}$$

$$p_{ij} = \iiint S_s N_i N_j \, dxdydz \tag{9b}$$

$$f_i = \iiint q N_i \, dxdydz \tag{9c}$$

Note that Eq. 9c is used when the discharge is distributed over a given volume. If the discharge is applied at a given node **i**, as is assumed in this study, f_i will be the actual discharge and Eq. 9c will not be used. Integration of Eq. 9 results in the following stiffness matrix for a tetrahedron:

$$[H]^e = \frac{1}{36V} \begin{bmatrix} k_x b_1^2 + k_y c_1^2 + k_z d_1^2 & k_x b_1 b_2 + k_y c_1 c_2 + k_z d_1 d_2 & k_x b_1 b_3 + k_y c_1 c_3 + k_z d_1 d_3 & k_x b_1 b_4 + k_y c_1 c_4 + k_z d_1 d_4 \\ & k_x b_2^2 + k_y c_2^2 + k_z d_2^2 & k_x b_2 b_3 + k_y c_2 c_3 + k_z d_2 d_3 & k_x b_2 b_4 + k_y c_2 c_4 + k_z d_2 d_4 \\ \text{Symmetry} & & k_x b_3^2 + k_y c_3^2 + k_z d_3^2 & k_x b_3 b_4 + k_y c_3 c_4 + k_z d_3 d_4 \\ & & & k_x b_4^2 + k_y c_4^2 + k_z d_4^2 \end{bmatrix} \tag{10a}$$

$$[P]^e = 6VS_s \begin{bmatrix} \frac{1}{60} & \frac{1}{120} & \frac{1}{120} & \frac{1}{120} \\ \frac{1}{120} & \frac{1}{60} & \frac{1}{120} & \frac{1}{120} \\ \frac{1}{120} & \frac{1}{120} & \frac{1}{60} & \frac{1}{120} \\ \frac{1}{120} & \frac{1}{120} & \frac{1}{120} & \frac{1}{60} \end{bmatrix} \tag{10b}$$

The stiffness matrix of the tetrahedral element, as shown by Eq. 10, are superimposed to obtain that of the triangular prism and the hexahedron.

Simultaneous Equations

After the stiffness matrix for each element is obtained, it can be superimposed to form the overall matrix for the entire aquifer, and the following simultaneous equations are obtained:

$$[H]\{s\} + [P]\{\partial s/\partial t\} - \{F\} = 0 \tag{11}$$

in which $[H]$ = a square matrix depending on permeabilities and element configurations; $[P]$ = a square matrix depending on specific storage and element configurations; $\{F\}$ = a column matrix representing the discharge or recharge at the nodes, positive for discharge and negative for recharge, $\{s\}$ = drawdown at the nodes; and t = time since pumping started. Using the implicit finite difference formula, Eq. 11 can be written as

$$[H]\left(\frac{\{s\}_t + \{s\}_{t-\Delta t}}{2}\right) + [P]\left(\frac{\{s\}_t - \{s\}_{t-\Delta t}}{\Delta t}\right) - \{F\} = 0$$

or

$$[C]\{s\}_t = [D]\{s\}_{t-\Delta t} + 2\{F\} \tag{12a}$$

in which

$$[C] = [H] + \frac{2}{\Delta t}[P] \tag{12b}$$

$$[D] = -[H] + \frac{2}{\Delta t}[P] \tag{12c}$$

Eq. 12 indicates that the drawdown at time t can be determined from that at time t - Δt. Because the drawdown at the initial time is everywhere zero, the drawdown at any time t can be obtained by dividing the time into several increments and determining the drawdown at the end of each time increment from Eq. 12. In addition to the requirements that the drawdown is zero or the discharge is prescribed at a specified number of points, the boundary condition which must be satisfied is that the sum of the nodal discharges along the well bore be equal to the constant pumping rate Q.

To simulate a real aquifer with irregular boundaries, a large number of finite elements or nodal points are generally needed, so Eq. 12 will consist of a large number of simultaneous equations. A serious problem in solving such a large number of equations is how to store the coefficients of $[C]$ matrix within the limited capacity of a computer. In the original computer program using cylindrical elements [Sonnenfeld, 1973; Huang and Sonnenfeld, 1974], the simultaneous equations were solved by the tridiagonal method [Ziewkiewicz and Cheung, 1967]. The aquifer was divided into two or more partitions. The coefficients of [C] matrix for each partition were computed one at a time and then stored on disks for later use. Consequently, the total number of storage locations for all equations was equal to that required by the largest partition only. It was later found that the tridiagonal method required considerable computer time because of the use of matrix inversions. The tridiagonal method was then replaced by a subroutine called SESOL (simultaneous equation solver) originally developed by Wilson, Bathe and Doherty [1974]. This subroutine also employes the method of partitions by storing the coefficients of [C] matrix into disks, one partition at a time. However, instead of matrix inversions, the Cholesky scheme of decomposing the [C] matrix into an upper and a lower triangular matrix was used. In addition to the savings in computer time, a major advantage of this subroutine is that there is practically no limit on the number of equations to be solved, as long as the computer has enough capacity to store the coefficients of two equations at a time. Nevertheless, it has the disadvantage that four disks are used, and the transfer of data in and out of disks may constitute a significant part of the computing cost.

The partition method is particularly useful for analyzing large aquifers having circumferential boundaries far away from the well. At the initial stage of pumping when the cone of depression extends over a limited area, it is not necessary to compute the drawdown in all partitions, because the drawdown outside the cone of depression is known to be zero. For the first time increment, only the drawdown in the first one or two partitions needs to be considered. As the time increases, more partitions should be added until all partitions are taken into consideration.

Boundary Conditions at Well

The first boundary condition, which must be satisfied at the well, is that the discharge from the well is constant and equals Q. A second condition, which must be met physically, is that the same drawdown exists for all nodal points at the well. For a fully penetrating well in a homogeneous aquifer, as was the case assumed by Theis [1935], the discharge is uniform along the well bore, so the column matrix $\{F\}$ in Eq. 12 can be determined by proportioning Q to all nodes at the well. The drawdown obtained from Eq. 12 will also be equal for those nodes at the well, so both conditions are satisfied. Unfortunately, this ideal situation does not exist in a partially penetrating well or a well surrounded by a nonhomogeneous or sloping aquifer. If the discharge is assumed uniform along the well bore, as was assumed by Hantush [1957] in the analysis of a partially penetrating well, the drawdown obtained from Eq. 12 will no longer be equal at the well, and the second boundary condition is not satisfied. Huang [1972] indicated that the assumption of a uniform discharge for a partially penetrating well would involve considerable error unless the point in question was at a large distance from the well. To satisfy both boundary conditions, the following special technique was developed.

First, a drawdown in the well is arbitrarily assumed, and the drawdown $\{s_1\}_t$ at all nodal points outside the well can be determined from Eq. 12, or

$$[C]\{s_1\}_t = [D]\{s\}_{t-\Delta t} + 2\{F\} \tag{13}$$

A second set of drawdown for all points outside the well, based on the same assumed drawdown in the well, is then determined by

$$[C]\{s_2\}_t = 0 \tag{14}$$

and it can easily be seen that the linear combination

$$\{s\}_t = \{s_1\}_t + \alpha\{s_2\}_t \tag{15}$$

is also a solution of Eq. 13, in which α is a constant and can be determined as follows:

When Eq. 12 is applied to each of the nodes at the well bore, using the assumed drawdown in the well and the resulting $\{s_1\}_t$ from Eq. 13, a discharge q is obtained for each node at the well, i. e.

$$\{q\} = \frac{1}{2}\left([C]\{s_1\}_t - [D]\{s\}_{t-\Delta t}\right) \tag{16}$$

By summing the discharges of all the nodes at the well bore, an equation of the following form is obtained:

$$\Sigma q = \frac{1}{2}\left([A]\{s_1\}_t - [D]\{s\}_{t-\Delta t}\right) \tag{17}$$

in which [A] and [B] are matrices with one row only. To satisfy the requirement that the well discharge is Q, $\{s\}_t$ must satisfy the equation

$$Q = \frac{1}{2} ([A] \{s\}_t - [D] \{s\}_{t - \Delta t}) \quad (18)$$

Subtracting Eq. 17 from 18

$$Q - \Sigma q = \frac{1}{2} [A] (\{s\}_t - \{s_1\}_t) \quad (19)$$

Combining Eqs. 15 and 19

$$\alpha = \frac{2(Q - \Sigma q)}{[A] \{s_2\}_t} \quad (20)$$

Knowing α, $\{s\}_t$ can be determined from Eq. 15.

Computer Program

The method described above was programmed for an IBM 360 computer, Model 65, at the University of Kentucky. The program can be used to determine the drawdown around an artesian well at various times since pumping started. The well may be infinitesimal or finite in radius with a uniform or nonuniform discharge along the well bore.

Figure 4 shows a hypothetical aquifer with irregular shape and cross section, which is divided into three-dimensional finite elements. To facilitate the automatic generation of data and simplify the program with practical applications in mind, the following assumptions are made:

(1) The initial piezometric surface is a plane, so the computed drawdown is independent of the actual values of the initial piezometric head. This assumption is quite practical and makes possible the use of drawdown as the only dependent variable regardless of initial piezometric heads. In fact, drawdown is one of the most important factors in groundwater investigations.

(2) The top and bottom boundaries of the aquifer can be approximated by a plane. This assumption eliminates the tedious work of reading in the coordinates of each nodal point. By simply specifying the coordinates of three points on each plane and the thickness of each layer along the center of well, the computer will divide the aquifer into layers and determine the nodal coordinates of each element.

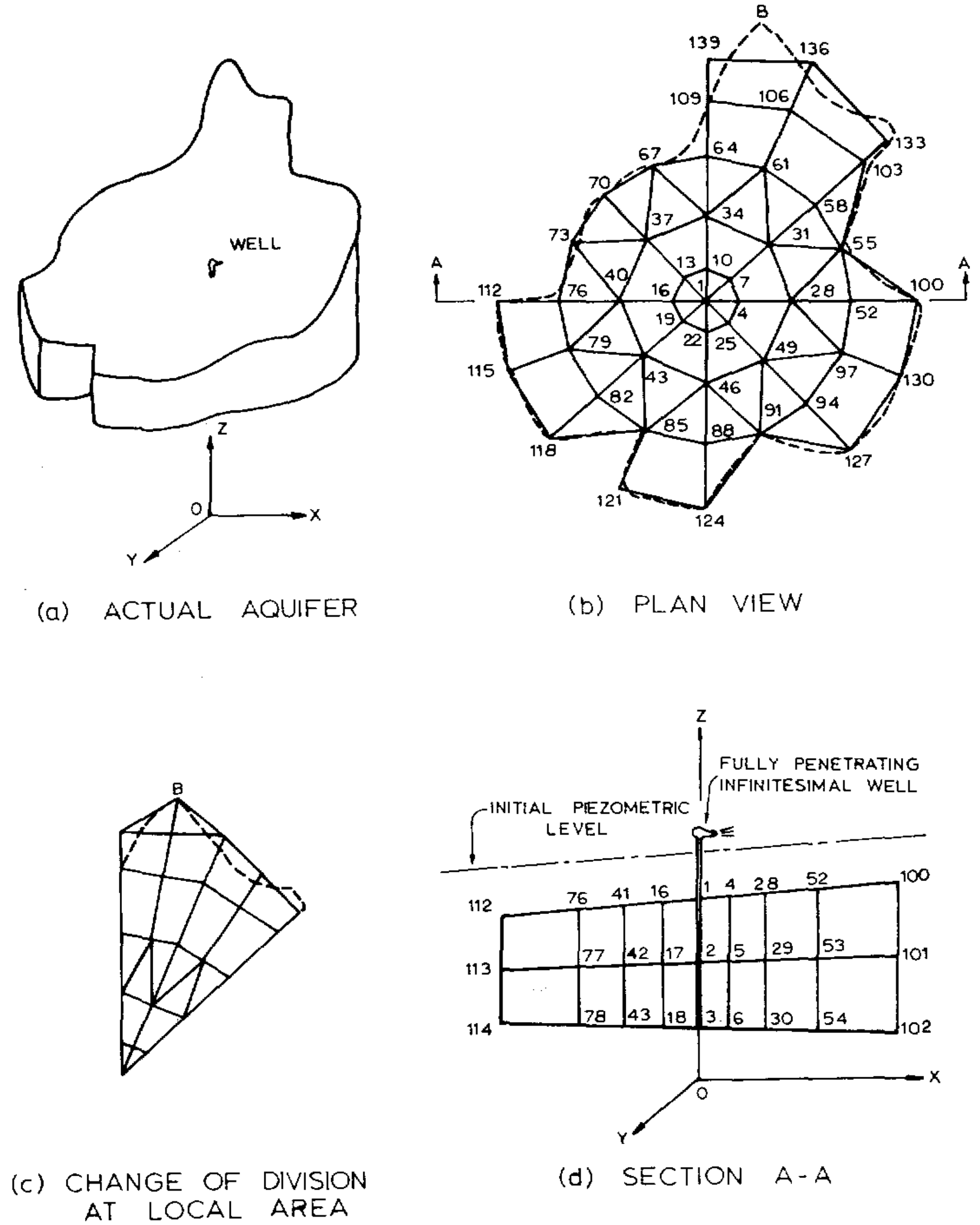


Figure 4. Division of aquifer into three-dimensional finite elements.

(3) The circumferential boundaries and the lateral surfaces of all finite elements are vertical. This assumption greatly facilitates the generation of nodal coordinates and the graphical presentation of the finite element network. Because each layer of finite elements has the same plan view, only one plotting is needed to show the plan configuration of finite elements.

(4) The three principal directions of permeability coincide with the global coordinate axes, x, y, and z. This assumption avoids the tedious process of assigning the three principal directions for each element and the transformation of stiffness matrix from local to global coordinates. In fact, most of the major aquifers are nearly horizontal with the direction of principal permeabilities in the vertical and the horizontal directions, which are in the direction of global coordinates.

(5) A zero drawdown or a constant discharge or recharge may exist at any given nodal point. This assumption makes possible the application of the program to aquifer evaluations, because discharges and recharges at various points can be assumed to simulate field conditions. These discharges or recharges are not those which result in a steady initial piezometric surface but are the additional discharges or recharges which take place after the pumping is started.

(6) The well is pumped at a constant rate of discharge. This assumption is for convenience only. The program can be easily modified to take variable discharge into account.

Figure 4 shows how an aquifer is divided into three-dimensional finite elements. To simulate the aquifer using the six-node and eight-node elements, first from the center of well a set of radial lines is drawn to divide the aquifer into a number of slices. Then, a set of centripetal polygonal prism, formed by the so-called loops, is constructed. The aquifer is divided into a number of layers. The thickness of each layer along the center of well, or z axis, is specified, and the thickness at other points along a vertical line is computed by proportion. For greater accuracy, it is desirable that the finite elements be as equilateral as possible. If some elements are too oblong, subdivisions can be made starting from the appropriate loop. Figure 4b shows that the subdivision starts from the fourth loop, counting the center of well as the first loop. If the well is finite in radius, instead of infinitesimal, the circumference of the well is at the second loop. Note that a partial radial line always exists between two full radial lines. If the boundary of aquifer lies on the fourth loop, an imaginary partial radial line still exists outside the boundary.

The division shown in Figure 4b does not match the boundary at point B. As the corner of aquifer must be on a radial line, one way to obtain a better match is to make a second subdivision starting from the fifth or sixth loop. When this decision is made, every fifth or sixth loop will be subdivided. This may not be desirable because the division at other part of the aquifer is satisfactory, and no more subdivision is needed. To change the division of one local area without affecting the remainder, the method shown in Figure 4c may be used. In this method, the partial radial line passing through nodes 61 and 136 is changed into a full radial line, and two partial radial lines are added starting from the fourth loop. In other words, when an aquifer is divided properly by a set of full and partial radial lines but one additional radial line is needed to match the boundary, instead of adding one radial line, the general rule is to add two lines; either changing a partial radial line to a full line and adding two partial lines one on each side of the full line, as shown in Figure 4c, or simply adding one full line and one partial line to keep the division in a regular pattern. This may indicate that, by a judicious selection of radial lines, any complex boundaries can be approximated by the subdivision method.

The ability to subdivide the aquifer into elements of approximately the same size will make possible the application of the computer program for aquifer evaluation. In this case, the drawdown in the vicinity of the well is not important, so larger elements can be used surrounding the well, and the discharge can be assumed uniform along the well bore. Discharges, recharges, or zero drawdown can be specified at various nodal points, and the drawdown at any point in the aquifer can be computed.

The numbering of nodal points is also shown in Figure 4. Starting from the center of well, the nodes are numbered first vertically from top to bottom, then tangentially counterclockwise starting from the positive x axis, and finally radially outward. The same rule is used to number the elements. When reading in the input data, it is not necessary for the user to determine the nodal or element number. Each node or element is located by a radial line number, a loop number, and a layer plane number. When these numbers are specified, the computer will determine the corresponding nodal or element number and use it in the computation. The computer can also plot out a plan view of the aquifer together with the drawdown at various points on a layer plane. Details of the computer program will be presented in the Appendix.

CHAPTER III

DATA AND RESULTS

Use of Dimensionless Parameters

To minimize the necessary work for preparing input data, it is convenient to use dimensionless parameters, especially when nonhomogeneous and anisotropic aquifers are involved. Instead of using directly the actual value of an input parameter, the ratio between the actual value and a basic value is specified. For a homogeneous and isotropic aquifer, the basic values of permeability and specific storage are their actual values, so a ratio of 1 is automatically assigned to all elements. For a nonhomogeneous aquifer, the permeability and specific storage which are the same for most elements are taken as the basic values, so only those which are different from the basic values need be read in as ratios to the basic values. The basic value for discharge is the discharge from the well. The discharge or recharge at any nodal point is specified as a ratio to the basic discharge. The basic value for distance is an arbitrary length L. Although L can be any length, a convenient way is to specify L as a unit distance, say 1 ft , so the actual distance in ft can be read in. The time is read in as a dimensionless ratio defined by

$$\tau = \frac{kt}{S_s L^2} \tag{21}$$

in which τ = dimensionless time, or time factor, t = actual time, k = basic permeability, S_s = basic specific storage, and L = basic length. In the computer program, a sequence of dimensionless times in logarithmic scale is arbitrarily assigned, and the actual time t, corresponding to each dimensionless time assigned, can be determined by

$$t = \frac{S_s L^2 \tau}{k} \tag{22}$$

When the permeabilities, specific storage, discharge, distance, and time are all read in as dimensionless ratios, the drawdown obtained from Eq. 12

will also be dimensionless and can be expressed as

$$\sigma' = \frac{k L s}{Q} \tag{23}$$

in which σ' = uncorrected dimensionless drawdown, and Q = basic discharge, or the discharge from the well. The actual drawdown can be determined by

$$s = \frac{Q \sigma'}{k L} \tag{24}$$

The dimensionless drawdown shown in Eq. 23 is based on an aquifer with a thickness b, instead of a unit thickness L. To determined the drawdown in an aquifer of unit thickness, Eq. 23 should be replaced by

$$\sigma = \frac{k b s}{Q} \tag{25}$$

in which σ = dimensionless drawdown, or drawdown factor, and b = thickness of aquifer along the centerline of well. Note that kb is defined as the transmissibility of an aquifer, and $S_s b$ as the storage coefficient. Therefore, Eqs. 21 and 25 can be written as

$$\tau = \frac{Tt}{S L^2} \tag{26a}$$

$$\sigma = \frac{T s}{Q} \tag{26b}$$

in which T = basic transmissibility, and S = basic storage coefficient. The dimensionless time and drawdown, as expressed by Eq. 26, were used previously in presenting numerical results and will be used again in this report.

Comparison with Theis Solution

Theis [1935] presented a method for determining the drawdown around a fully penetrating artesian well. By assuming that the aquifer is homogeneous, isotropic, infinite in areal extent, and uniform in thickness and that the well is infinitesimal in radius, the equation for drawdown can be expressed as

$$s = \frac{Q}{4\pi T} W(u) \tag{27a}$$

$$u = \frac{r^2 S}{4Tt} \tag{27b}$$

in which W(u) = well function = $\int_u^\infty \frac{e^{-x}}{x}\,dx$, and r = radial distance from the well to a point at which drawdown is to be determined. Values of W(u) for values of u were tabulated by Wenzel and reproduced by Dewiest [1965].

Using the dimensionless factors defined by Eq. 26, Eq. 27 can be written as

$$\sigma = \frac{1}{4\pi} W(u) \tag{28a}$$

$$u = \frac{\left(\frac{r}{L}\right)^2}{4\tau} \tag{28b}$$

For any given time factor, τ, the corresponding drawdown factor, σ, at any given radial distance, r, can be determined from Eq. 28.

Figure 5 gives a comparison between Theis and finite element solutions. Theis solution, which is obtained from Eq. 28, is indicated by the solid curves; while the finite element solution is indicated by the individual circles. It can be seen that the finite element solution checks closely with Theis solution, thus the validity of the finite element formulation is verified.

In the finite element analysis, a large aquifer having a circumferential boundary at a distance of 148L from the well is employed. Because the flow is axisymmetric, only a 30° slice is considered, as shown in Figure 6. The slice is subdivided twice, one starting from loop 11 and the other from loop 14. Because of full penetration, only one layer of elements is used. The aquifer is divided into 30 elements with a total of 86 nodes. The drawdowns are computed at fifteen dimensionless time increments, i.e. at the end of 0.5, 1, 2, 5, 10, 20, 50, 100, 200, 500, 1000, 2000, 5000, and 10,000. These increments are also used for all cases reported herein.

Aquifer of Variable Thickness

Figure 7 shows the cross section for an aquifer of variable thickness and its division into finite elements. The plan view of the aquifer is exactly the same as that shown in Figure 6, but the top and bottom boundary planes converge toward the well instead of being horizontal. To investigate the variation

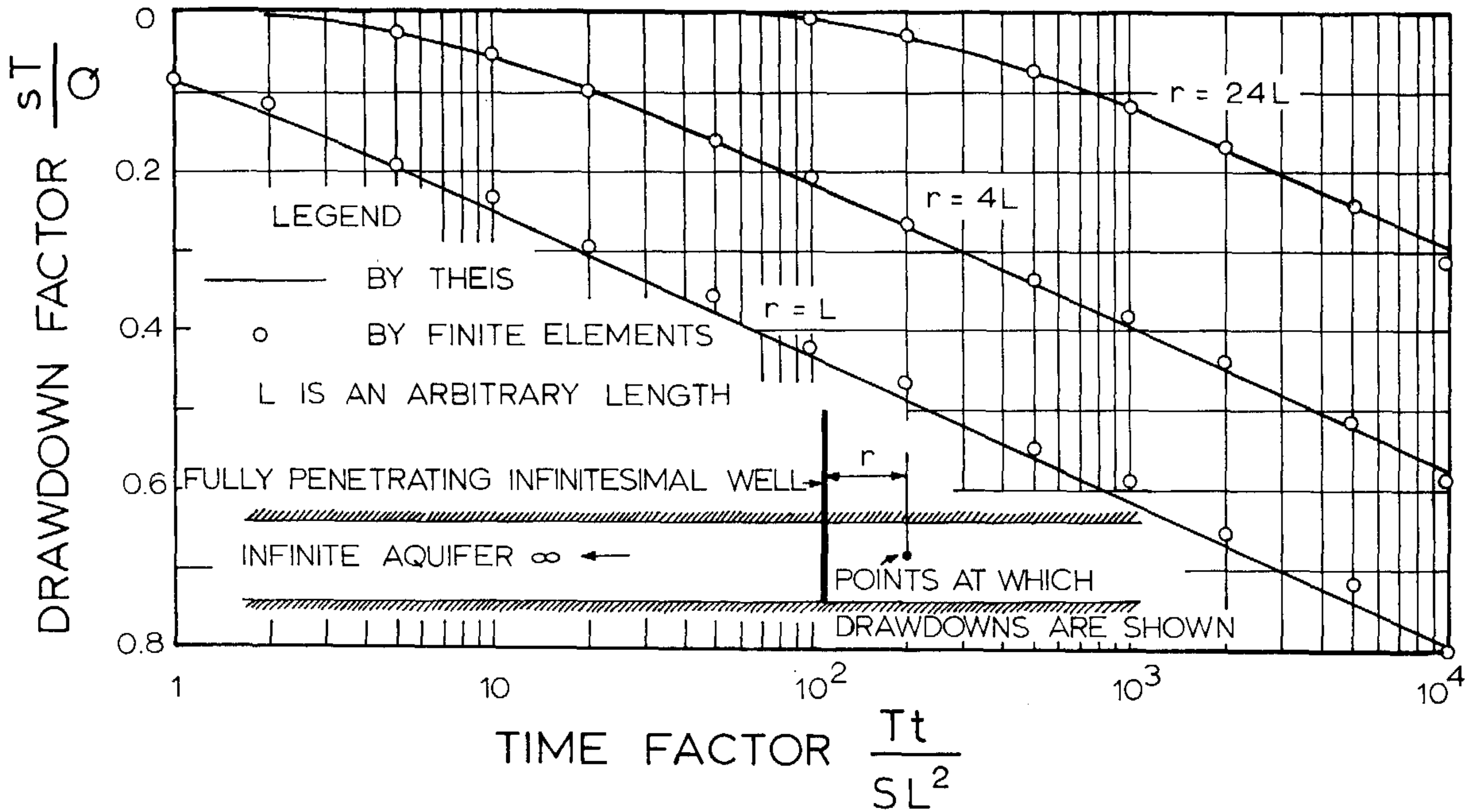


Figure 5. Comparison between finite-element and Theis solutions.

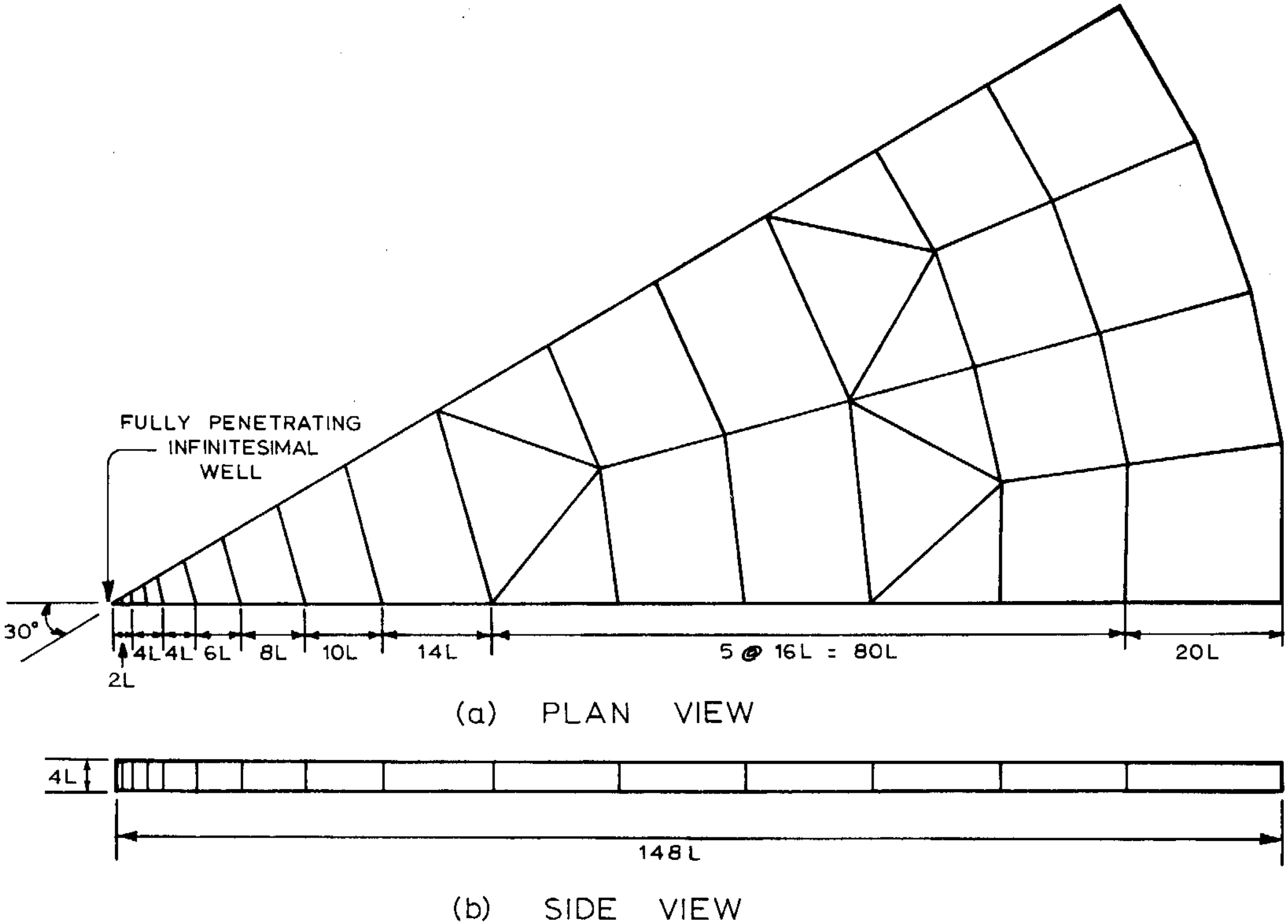


Figure 6. Simulation of an axisymmetric aquifer by a partial aquifer.

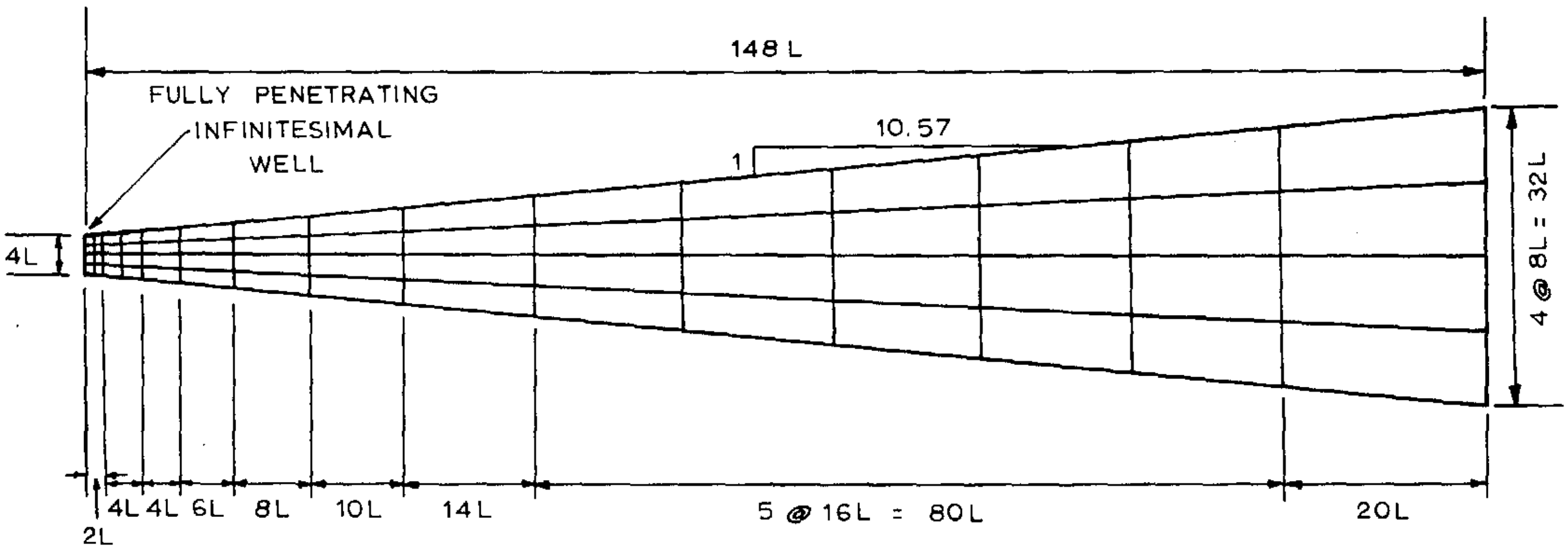


Figure 7. Cross section of aquifer with variable thickness.

of drawdown with depth, four layers are used. Theoretically, it is only necessary to consider the top or bottom half of the aquifer, because symmetry of the cross section indicates no flow through the horizontal plane at the mid height, where an impervious boundary can be assumed. However, all four layers are used because the purpose of this example is to check the correctness of the computer program and find whether a symmetrical solution can be obtained. It was found that the computed drawdowns are all symmetrical with respect to the mid height, thus the correctness of the computer program is further confirmed.

Figure 8 shows a comparison between the solution based on an aquifer of uniform thickness and that of variable thickness. In both cases, the thickness of aquifer is the same along the axis of the well, so the transmissibility and storage coefficient of both aquifers are identical. The solution based on uniform thickness is represented by circles; while that based on variable thickness is represented by crosses, when the point is located at the mid height, and by triangles, when it is on the top or bottom boundary. To give a continuous picture on the change in drawdown with time, the circles, crosses, and triangles are connected by straight lines. It can be seen that the plotted points do not form a smooth curve because of the rather large time increments employed.

Figure 8 shows that at short pumping times the drawdown based on uniform thickness is not too much different from that based on variable thickness. This is reasonable because the cone of depression at short times extends only to a limited distance from the well, where the change in thickness is not significant. As the pumping time increases, the drawdown based on variable thickness does not increase as rapidly as that based on uniform thickness. This is also reasonable because the greater the thickness, the more the water released from storage, and the less the drawdown.

The fact that in the case of variable thickness the drawdown at the mid height is greater than that at the top or bottom boundary can also be explained by theory. If a flow net is drawn on the cross section of the aquifer, the equipotential lines will form a set of curves perpendicular to both the top and bottom boundaries. Consequently, if a vertical line is drawn tangent to an equipotential line at the midheight, the piezometric head along the vertical line will increase,

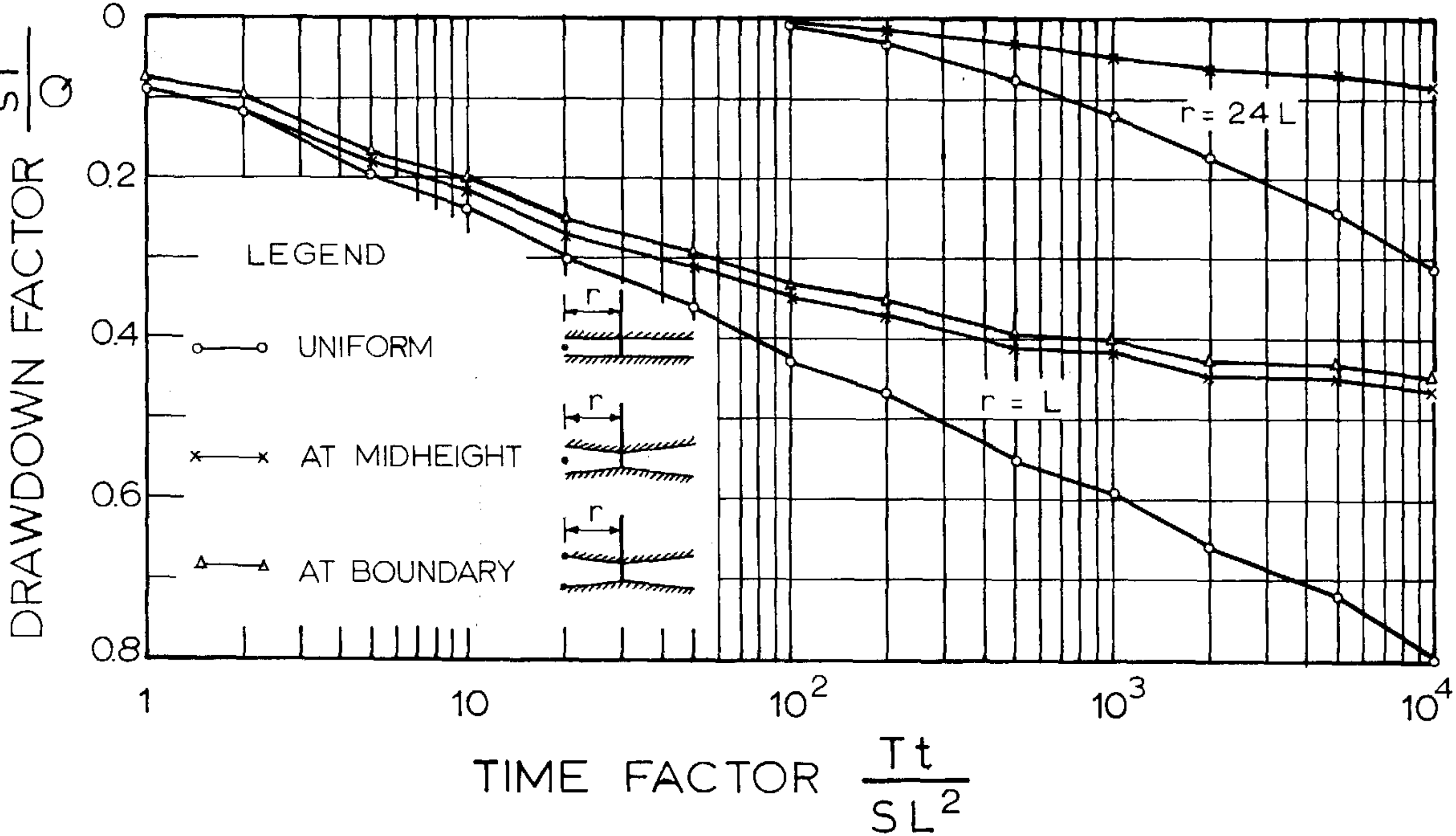


Figure 8. Comparison of solutions: uniform thickness versus variable thickness.

or the drawdown decreases, as the point moves toward both boundaries.

Comparison with Cylindrical Elements

Figure 9 shows a hypothetical problem involving a partially penetrating infinitesimal well in a nonhomogeneous aquifer, which was solved previously by Huang and Sonnenfeld [1974] using cylindrical finite elements. The transmissibility in the shaded region is assumed to be four times greater than that in the unshaded region, and the storage coefficient is assumed to be two times greater. It can be seen that the cylindrical elements do not match with either the boundary of the aquifer or the boundary between the two regions, which are indicated by the dotted lines. Because the present program can match the boundary more closely, it will be interesting to compare the solution obtained by the present program with that by the cylindrical elements.

Figure 10 shows a computer plot of the plan view and the dimensionless drawdown on the top boundary at a dimensionless time of 10,000. The six- and eight-node elements match exactly all boundaries. The thickness of each layer is the same as that shown in Figure 9. The aquifer is divided into 432 elements with a total of 495 nodes, in contrast to the 380 elements and 495 nodes used previously for cylindrical elements. Note that, if space is available, the drawdown is printed directly on each element to represent the value at the centroid of the element, which is marked by a cross. Otherwise, an alphabetical character is printed inside the element, and the corresponding drawdown is printed on the right side of the aquifer. Although no alphabetical letters are printed within those elements surrounding the well, they can be identified by the rule that the first element, or element A, is marked by a cross and other elements are counted counterclockwise in alphabetical order. The drawdown listed at the bottom is not the original plotting. The original plotting was on the right but was retyped to conserve space.

Figure 11 gives a comparison of solutions between cylindrical elements and six- and eight-node elements. The solid curves, which were presented previously by Huang and Sonnenfeld [1974], are the solution based on cylindrical elements; while the individual circles represent the solution based on six- and

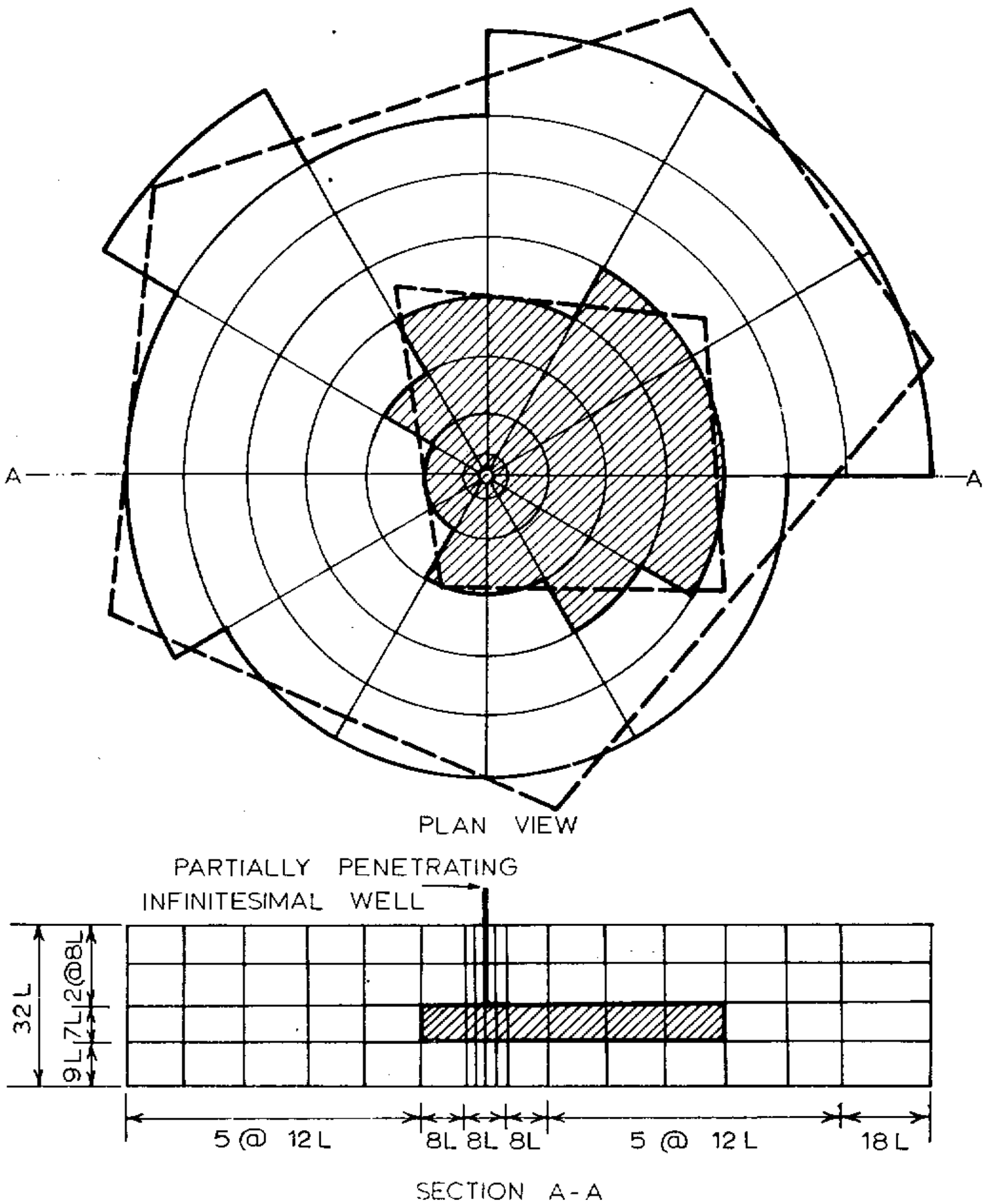


Figure 9. Approximation of a nonhomogeneous aquifer by cylindrical elements.

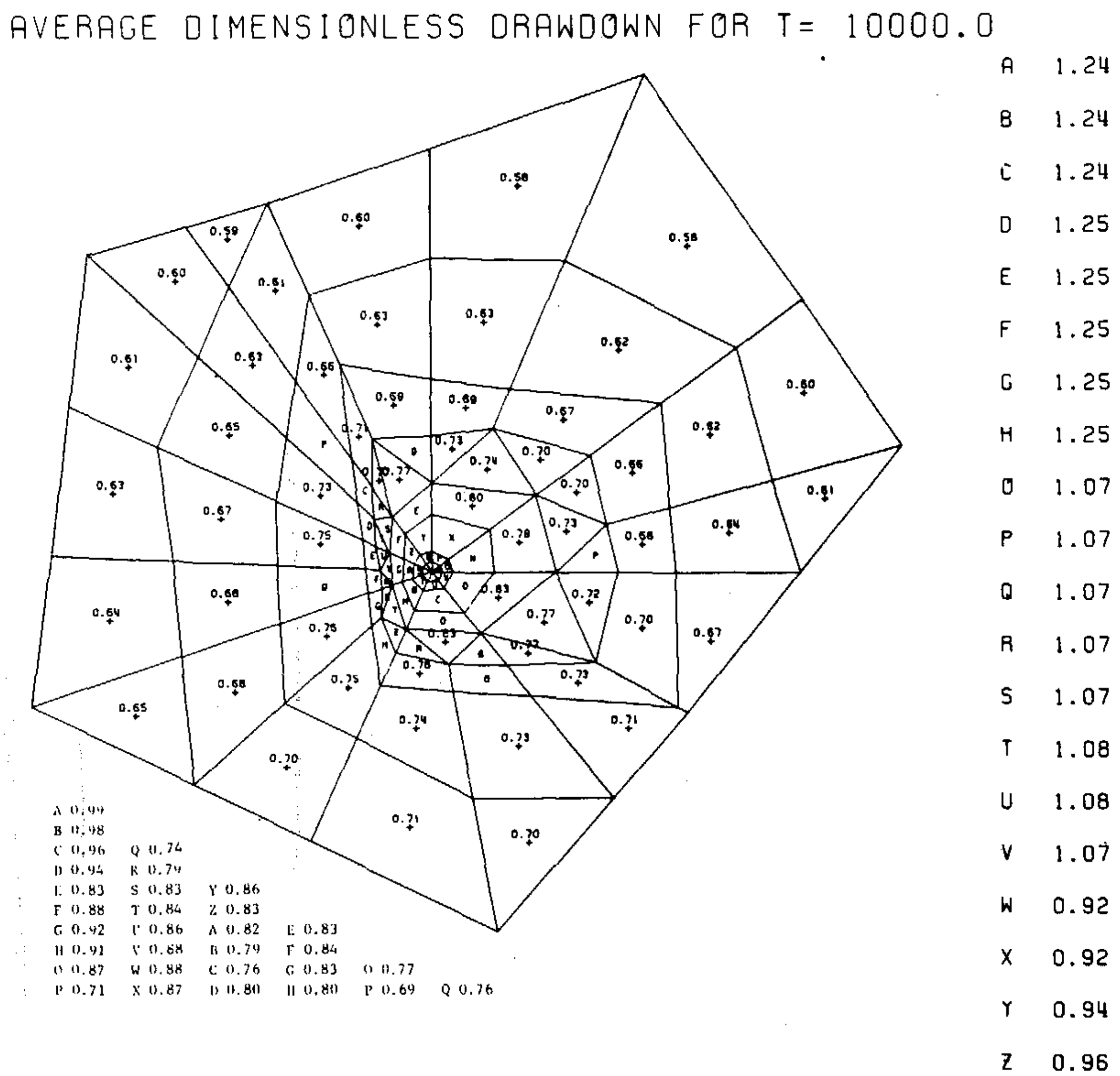


Figure 10. Exact matching of boundaries by six- and eight-node elements.

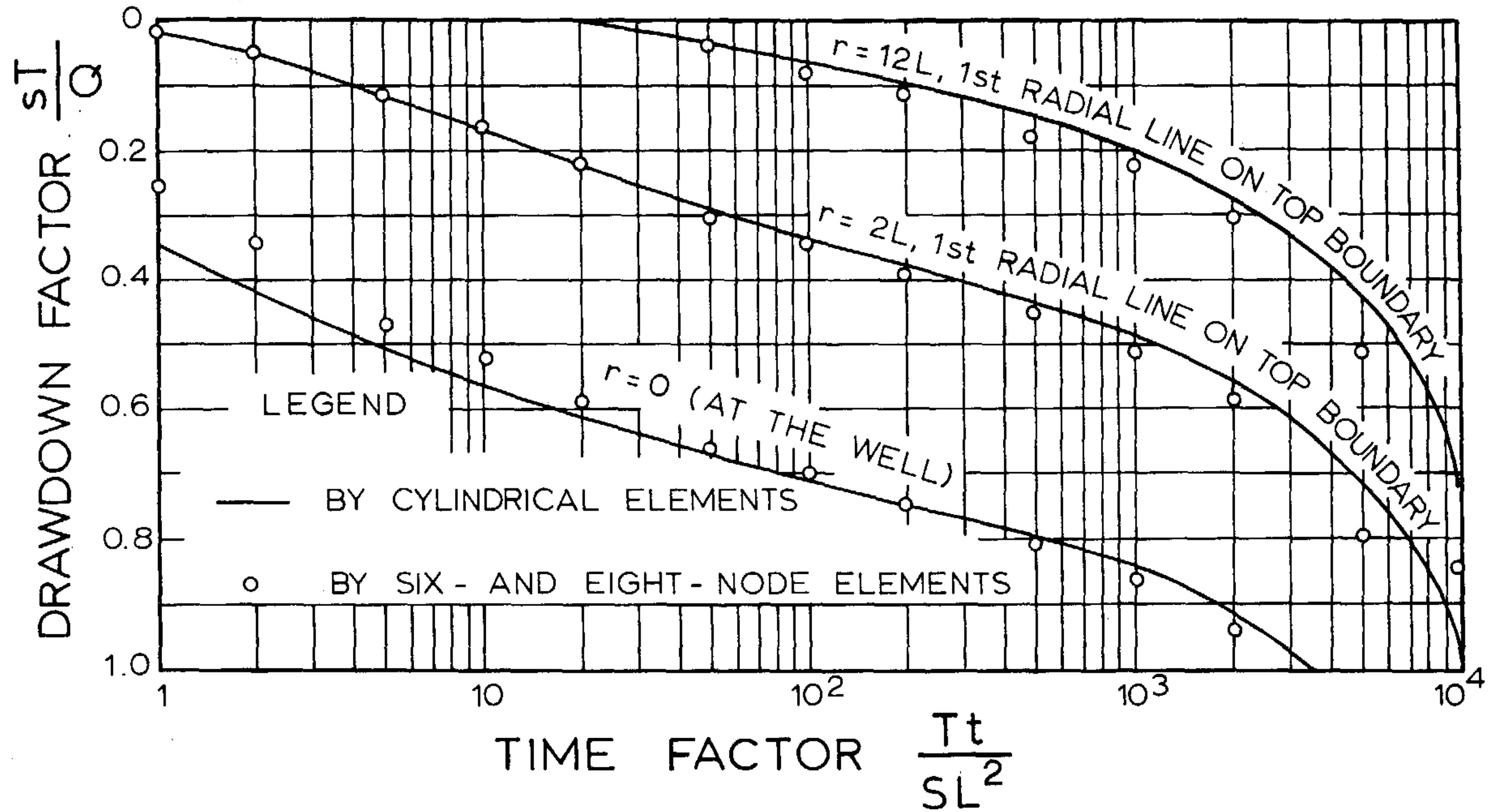


Figure 11. Comparison of solutions: cylindrical elements versus six- and eight- node elements.

eight-node elements. The drawdown and time are expressed in terms of the transmissibility and storage coefficient of the unshaded region. It can be seen that the difference in boundaries only causes a small discrepency in the computed drawdown. The good agreement between the two methods, which are completely different not only in the formation of stiffness matrix but also in the solution of simultaneous equations, affords further proof that the computer program developed in this research is theoretically correct.

Effect of Finite Element Division

Figure 12 shows the same aquifer which is subdivided in a different way. The figure, which presents the dimensionless drawdown on the top boundary at a dimensionless time of 10,000, was plotted by the computer except that short lines were added later to indicate the shaded area. The six- and eight-node elements match exactly the circumferential boundary but not the boundary between the two regions. The aquifer is divided into 288 elements with a total of 280 nodes, whereas 432 elements and 495 nodes are used in Figure 10.

Figure 13 shows the effect of finite element divisions on the computed cone of depression at dimensionless times of 100, 1000, 5000, and 10,000. The cone of depression is measured along line A-A on the top boundary of the aquifer. The solid lines are the solution when the aquifer is divided according to Figure 12; while the individual circles represent the solution when divided according to Figure 10. Although Figure 12 employs only four triangular prisms surrounding the well, the solution is not too much different from Figure 10 where eight prisms are used. This may indicate that approximate results can be obtained even a small number of elements is employed. Note that both solutions check closely on the right half of Figure 13 but not as close on the left half, because the number of elements used on the left half is much greater in Figure 10 as compared to Figure 12.

Aquifer with Arbitrary Boundaries

To illustrate the applicability of the computer program to a nonhomogeneous aquifer with arbitrary top and bottom boundaries, consider again the aquifer shown in Figure 9 except that the top and bottom boundaries are not

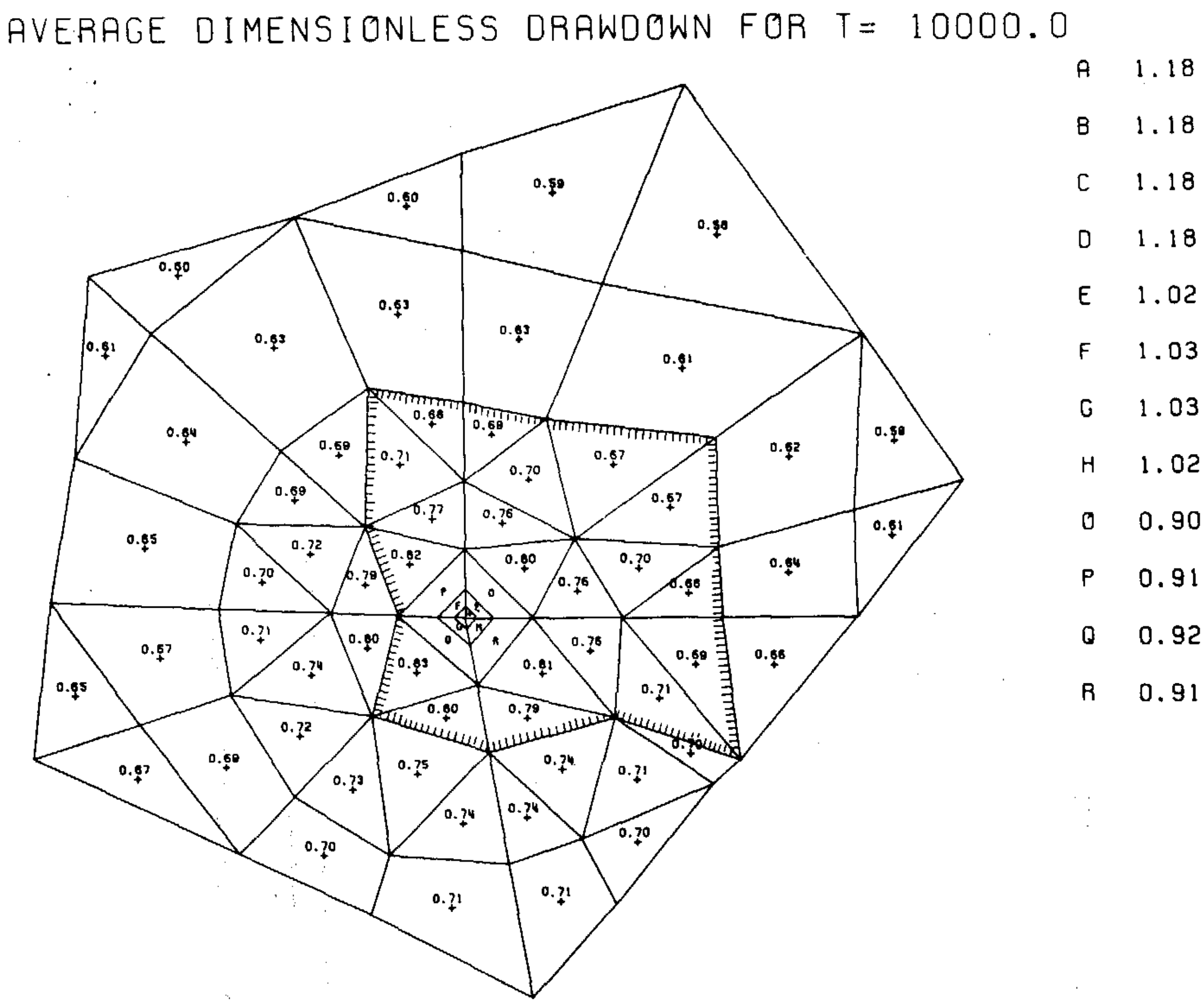


Figure 12. Approximation of a nonhomogeneous aquifer by six- and eight-node elements.

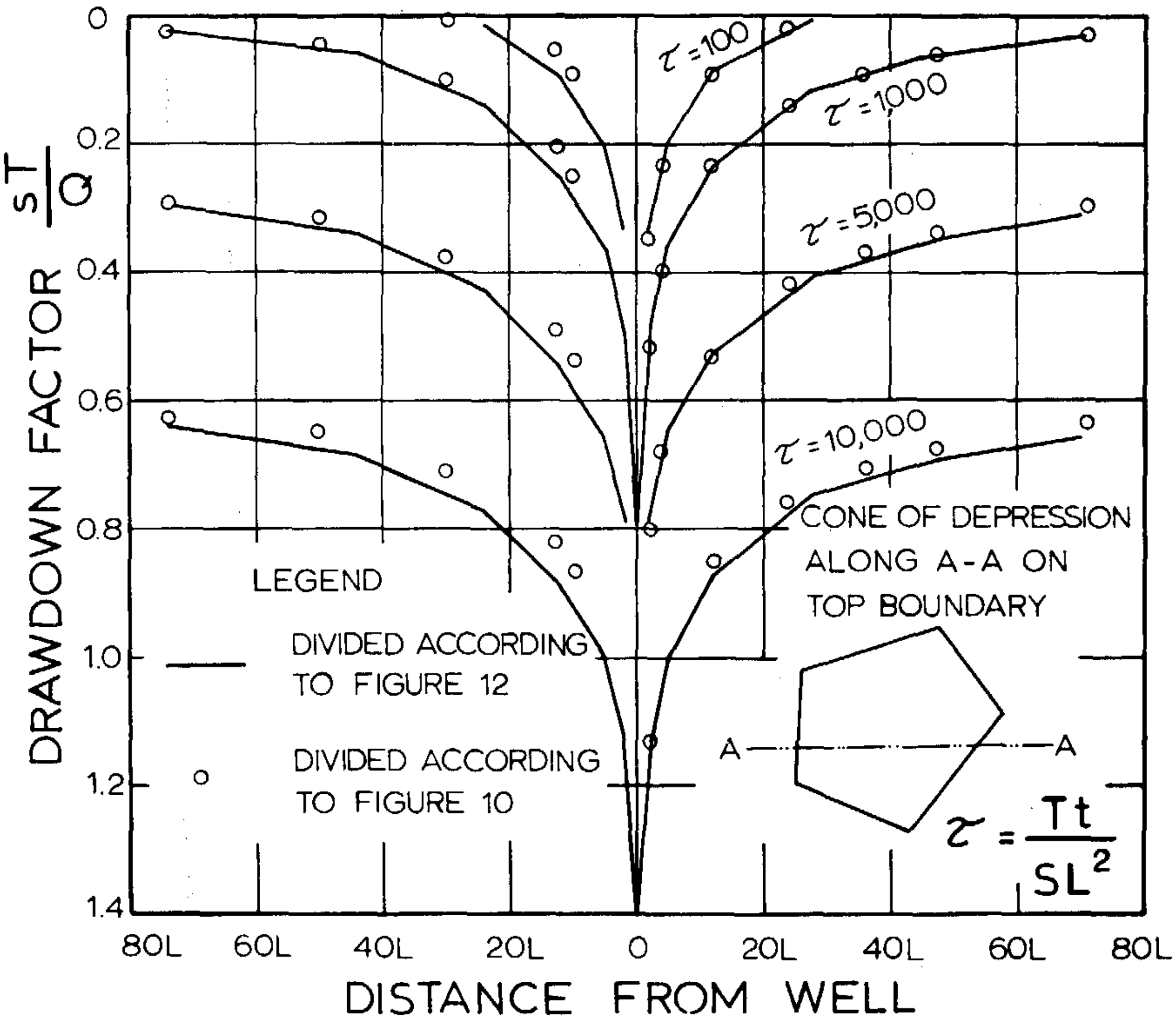


Figure 13. Effect of finite element division on drawdown.

horizontal. The plan view of the finite element subdivision is exactly the same as that shown in Figure 12, and the cross section along line A-A is shown in Figure 14. On each vertical line the aquifer is divided into layers of equal thickness. The region with a higher permeability and a greater storage coefficient is indicated by the shaded area. The thickness of the aquifer at the location of the well is 32L, which is the same as that of the horizontal aquifer, and the thickness does not change in a direction perpendicular to line A-A.

Figure 15 shows the cone of depression of the sloping aquifer as compared to that of the horizontal aquifer. The drawdown of the horizontal aquifer is indicated by solid lines and that of the sloping aquifer by dotted lines. It can be seen that the difference in drawdown between the two aquifers is greater on the right side of the well than on the left side. This is reasonable because the sloping aquifer has a much larger volume on the right side, which releases more water from storage thus reduces the drawdown.

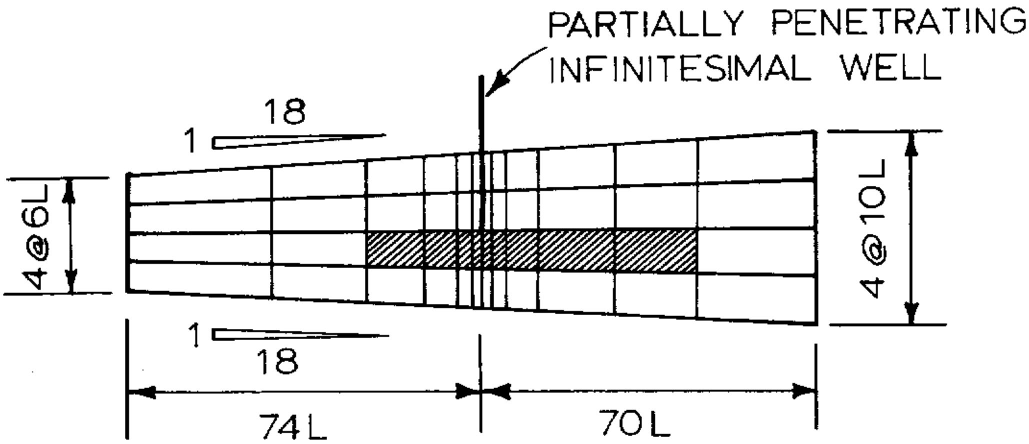


Figure 14. Cross section of a sloping aquifer.

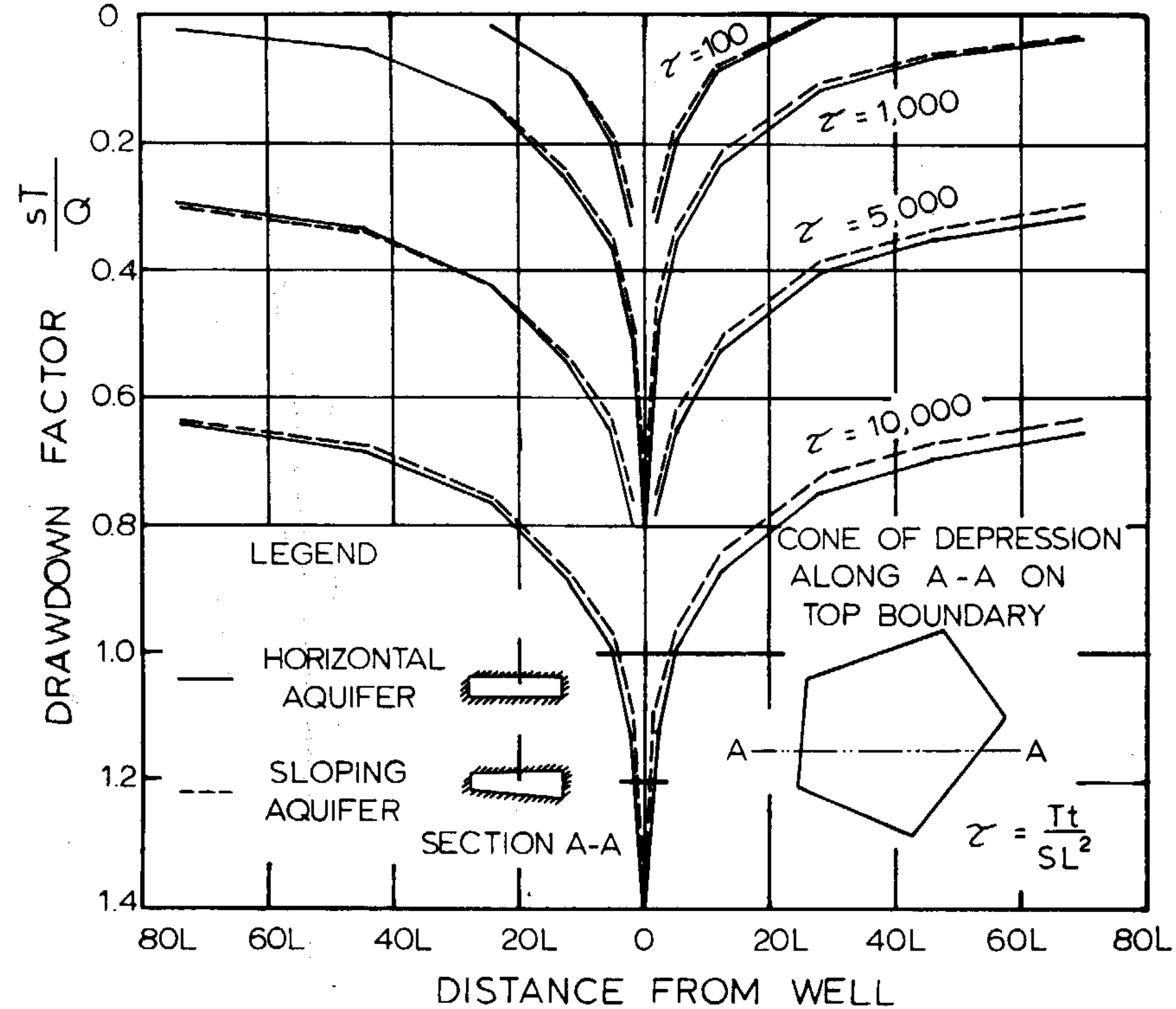


Figure 15. Comparison of solutions: horizontal aquifer versus sloping aquifer.

CHAPTER IV

SUMMARY AND CONCLUSIONS

A computer program based on three-dimensional finite elements was developed for analyzing unsteady flow toward artesian wells. The program can be used for determining the drawdown around a partially penetrating artesian well in a nonhomogeneous and anisotropic aquifer of irregular shape and cross section. The well may be assumed infinitesimal or finite in radius. In the case of an infinitesimal well, some computer time can be saved by assuming that the discharge is uniformly distributed along the well. This assumption does not satisfy the boundary condition at the well and may yield inaccurate results in the vicinity of the well. Therefore, an option is provided which satisfies the boundary condition without assuming a uniform distribution of discharge. The program can also be applied to a group of wells or for aquifer evaluation by specifying appropriate discharges or recharges at various points in the aquifer. The program was well documented and can be used by engineers and scientists for solving complex problems encountered in practice.

The use of the computer program requires a minimum amount of input data. By assuming the top and bottom boundaries of the aquifer as two arbitrary planes, the program will divide the aquifer into six- or eight-node elements and generate their nodal coordinates automatically. To facilitate the use of the program, it is not necessary for the user to number the nodes or elements. Every node or element is located by three numbers; viz. a radial line number, a loop number, and a layer plane number, which can be easily identified. Due to the use of disks for intermediate storage, there is practically no restriction on the size of aquifer or the number of elements employed. If the number of simultaneous equations is large, they should be divided into a sufficient number of partitions to keep the storage within the capacity of the computer.

The results of this study indicate that unsteady flow toward artesian wells can be analyzed effectively by three-dimensional finite elements. A comparison between the finite element solution by the computer program and the exact mathematical solution by Theis shows that both solutions check closely. The solution obtained from the computer program for a nonhomogeneous aquifer was checked against that obtained previously by the use of cylindrical elements, and both are found in good agreements. The program was applied to a variety of cases, and reasonable results were obtained.

REFERENCES

Desai, C. S., and J. F. Abel, Introduction to the Finite Element Method, pp. 380-403, Van Nostrand Reinhold, New York, 1972.

De Wiest, R. J. M., Geohydrology, pp. 352-359, John Wiley, New York, 1965.

Hantush, M. S., Nonsteady flow to a well partially penetrating an infinite leaky aquifer, Proc. Iraqi Sci. Soc., 1(10), 10-19, 1957.

Huang, Y. H., Finite element analysis of flow toward artesian well, J. Hydraul. Div. Amer. Soc. Civil Eng., 98 (HY11), 2055-2059, 1972.

Huang, Y. H., Unsteady flow toward an artesian well, Water Resour. Res., 9(2), 426-433, 1973.

Huang, Y. H., and S. L. Sonnenfeld, Analysis of unsteady flow toward an artesian well by three-dimensional finite elements, Water Resour. Res., 10(3), 591-596, 1974.

Javendel, I., and P. A. Witherspoon, Application of the finite element method to transient flow in porous media, Soc. Petrol. Eng. J., 8(3), 241-252, 1968.

Javandel, I., and P. A. Witherspoon, A method of analyzing transient fluid flow in multilayered aquifers, Water Resour. Res., 5(4), 856-869, 1969.

Pinder, G. F., and E. O. Frind, Application of Galerkin's procedure to aquifer analysis, Water Resour. Res., 8(1), 108-120, 1972.

Remson, I., G. M. Hornberger, and F. J. Molz, Numerical Methods in Subsurface Hydrology, pp. 322-336, John Wiley, New York, 1971.

Sonnenfeld, S. L., Three Dimensional Finite Element Analysis of Unsteady Flow Toward An Artesian Well Using Cylindrical Coordinates, Master's Thesis, University of Kentucky, Lexington, Ky., 1973.

Theis, C. V., The relation between the lowering of the piezometric surface and the rate and duration of discharge of a well using a groundwater storage, Trans. Amer. Geophys. Union, pp. 519-524, 1935.

Wilson, E. L., K. Bathe, and W. P. Doherty, Direct solution of large systems of linear equations, Computers and Structures, 4(2), pp. 363-372, 1974.

Zienkiewicz, O. C., and Y. K. Cheung, The Finite Element Method in Structural and Continuum Mechanics, pp. 148-168, McGraw-Hill, New York, 1967.

APPENDIX COMPUTER PROGRAM

PART I PROGRAM INFORMATION

1.1 Origin of Program

This computer program by the code name of UNFLAW (UNsteady FLow to Artesian Wells) was developed by Yang H. Huang and Shen-Jyh Wu, Department of Civil Engineering, University of Kentucky. The program is the end product of a research project entitled "Analysis of Unsteady Flow Toward Artesian Wells by Three Dimensional Finite Elements" supported partially by the Office of Water Resources Research, Department of the Interior. The subroutine for solving simultaneous equations was provided by Dr. E. L. Wilson, University of California at Berkeley. Technical assistances on the use of IBM 360 computer and Calcomp plotter were provided by the University of Kentucky Computing Center. These supports are gratefully acknowledged.

1.2 Purpose of Program

Although certain analytical and numerical methods are available for determining the drawdown around an artesian well, most of them are based on many simplifying assumptions. Some of these assumptions are that the aquifer is homogeneous, isotropic, two-dimensional, horizontal in position, uniform in thickness, and infinite in areal extent, that the well is infinitesimal in radius, and that the discharge is uniform along the well bore. As none of these assumptions are strictly valid for an artesian well in an actual aquifer, the development of a general computer program capable of taking the actual conditions into account is needed.

The purpose of this program is to provide a three-dimensional finite element method for determining the drawdown around an artesian well. The program considers a variety of cases possibly encountered in the field. Following are the cases to which the program can be applied: (1) the top and bottom boundaries of the aquifer can be planes oriented at any given direction, but the circumferential boundary of the aquifer has to be vertical; (2) the aquifer may be homogeneous or nonhomogeneous, isotropic or anisotropic; (3) the well may be infinitesimal or finite in radius, fully or partially penetrating; (4) the discharge along the well bore may be assumed uniform or nonuniform; and (5) discharge, recharge or zero drawdown can be specified at any given point. A list of these cases is shown in Figure A 1.

1.3 Problem Statement

When a well is drilled into a confined aquifer, the water level in the well will rise to an initial elevation. When water is pumped from the well, the water level in the well will be lowered; the difference between the initial and the lowered elevation is called the drawdown. The drawdown occurs not only in the pumping well but also in the aquifer; the latter can be measured by installing observation wells or piezometers at various points in the aquifer. This program was written for determining theoretically the drawdown in the well and in the aquifer at various times since pumping started. Information on the drawdown in the well can be used to determine the type of pumps required, while that in the aquifer can be used to predict ground settlements and thus prevent overdraft.

Although pumping tests can furnish valuable information on drawdown, they are quite expensive and time-consuming and cannot be used for routine design purposes. Once the formation constants and boundary conditions of aquifers are known in a given region through pumping tests and subsurface investigations, the computer program will determine the drawdown due to a pumping well located anywhere in the region. The ability to predict the drawdown before the well is installed will make it possible to search for the optimum location, so that enough water can be obtained from the well at a minimum cost.

1.4 Areas of Application

This computer program is intended to solve part of an overall problem on groundwater simulation or modelling. Although the program was written for a single artesian well and yields very accurate results on the drawdown in the vicinity of the well, it can also be used directly for aquifer simulation or evaluation. Knowing the aquifer properties, hydrogeologic boundaries, and the discharge or recharge at various nodal points, the time variation of water level in the aquifer can be determined. Information obtained from this program will be

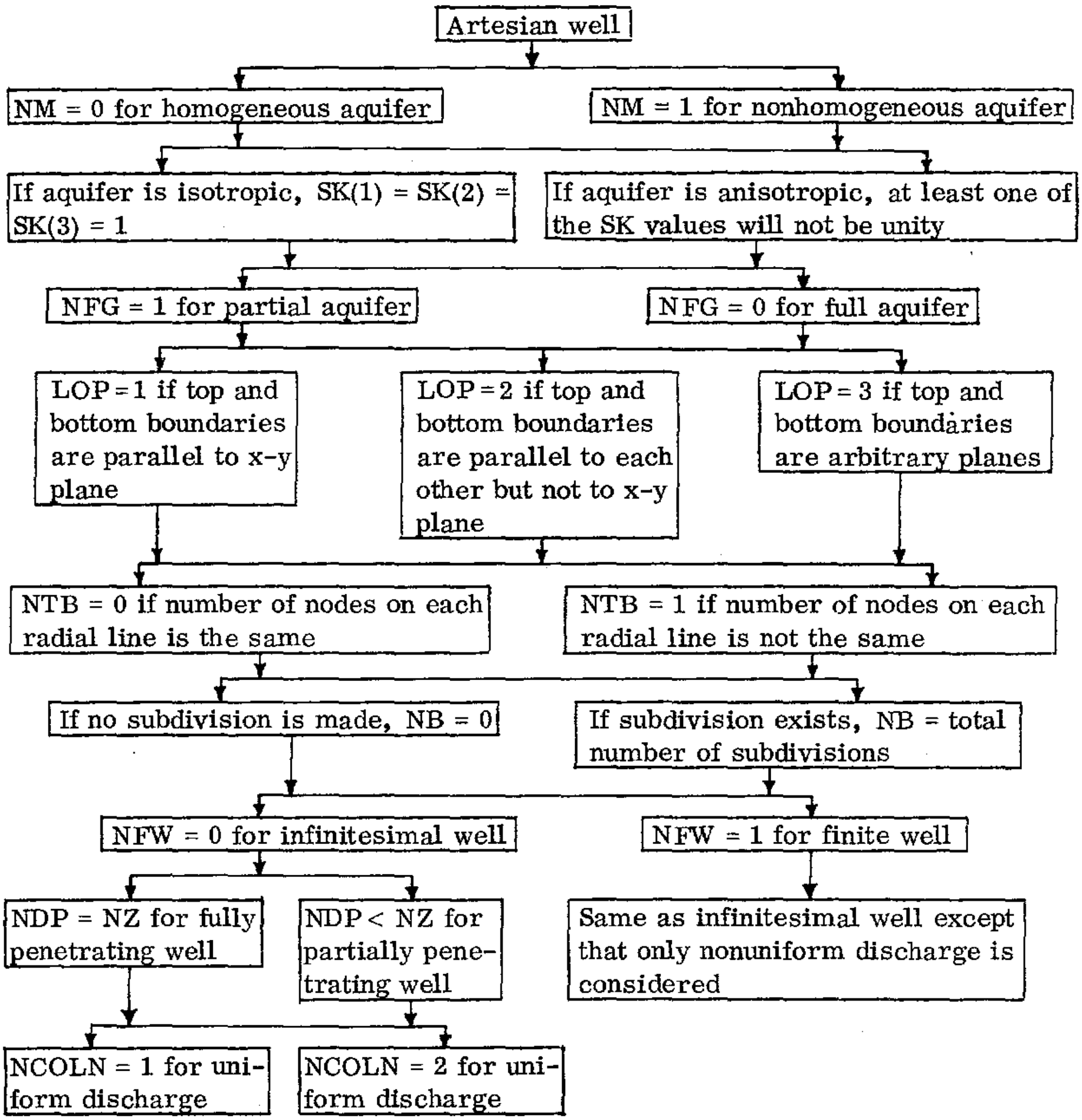


Figure A1. Cases considered by computer program.

of great value to the management, control and protection of groundwater resources.

1.5 Methods of Computation

The methods of computation were described in Chapter II of this report. Only these methods not described previously will be reported here.

1.5a Use of Local Coordinates

Eqs. 6 and 7, which are used for determining the stiffness matrix, are based on global coordinates, x, y and z. However, in actual calculations the origin of the coordinates is moved to the centroid of each element. The reason for this change is to avoid the multiplication of very large numbers when an infinite aquifer with large values of x, y and z is involved. The global coordinates in Eqs. 6 and 7 are replaced by the local coordinates, x', y' and z', expressed as

$$\begin{aligned} x' &= x - x_c \\ y' &= y - y_c \\ z' &= z - z_c \end{aligned} \qquad \text{(A1)}$$

in which $x_c = (x_1 + x_2 + x_3 + x_4)/4$, $y_c = (y_1 + y_2 + y_3 + y_4)/4$, and $z_c = (z_1 + z_2 + z_3 + z_4)/4$.

1.5b Equations for Top and Bottom Boundaries

The top and bottom boundaries of an actual aquifer may not be a plane, but they can be approximated by a plane. Each plane is defined if the coordinates of three points on the plane are given. Assume that the boundary can be approximated by the following linear equation

$$A_1 x + A_2 y + A_3 z = B \qquad \text{(A2)}$$

If the coordinates of three points, $P_1(x_1, y_1, z_1)$, $P_2(x_2, y_2, z_2)$, and $P_3(x_3, y_3, z_3)$, are given, the coefficients of Eq. A2 can be obtained by

$$\begin{aligned} A_1 &= (y_1 - y_2)(z_1 - z_3) - (y_1 - y_3)(z_1 - z_2) \\ A_2 &= (z_1 - z_2)(x_1 - x_3) - (z_1 - z_3)(x_1 - x_2) \\ A_3 &= (x_1 - x_2)(y_1 - y_3) - (x_1 - x_3)(y_1 - y_2) \end{aligned} \qquad \text{(A3)}$$

and $B = A_1x_1 + A_2y_1 + A_3z_1$

Eq. A3 is used only when the top and bottom boundaries are not horizontal. If both boundaries are horizontal, or parallel to the x-y plane, only the thickness of aquifer needs be specified, and Eq. A3 is not used.

1.5c Average Drawdown for Plotting Purpose

To give a more clear picture on the distribution of drawdown on a given layer plane, the average drawdown at the centroid of each triangle or quadrilateral, instead of at each node, is plotted. After the nodal drawdowns at the end of each time interval are computed, the average drawdown is determined by

$$s_a = \frac{\sum_{i=1}^{n} s_i}{n} \tag{A4}$$

in which s_a = average drawdown, s_i = drawdown at node i, and n = 3 for six-node elements and 4 for eight-node elements. This average drawdown is assumed to exist at the centroid of each triangle or quadrilateral, the coordinates of which can be determined by

$$x_c = \frac{\sum_{i=1}^{n} x_i}{n} \qquad y_c = \frac{\sum_{i=1}^{n} y_i}{n} \tag{A5}$$

in which x_c and y_c = coordinates of centroid, and x_i and y_i = coordinates of nodal points.

1.5d Dimensional Units

The program does not contain any conversion constants. Any units, either in British or SI system, can be used as long as they are consistent. If the distance is in feet and the time in seconds, then the discharge must be expressed in cubic feet per second, and the permeability in feet per second. If the distance is in meters and the time in minutes, then the discharge must be expressed in cubic meters per minute, and the permeability in meters per minute. The U. S. hydrological units cannot be used.

As has been described in Chapter III, dimensionless ratios are used as data input. The output can be either dimensionless or dimensional, depending on the desire of the user. To obtain the dimensionless drawdown as defined by Eq. 26b, the basic values of permeability, specific storage, discharge, and length must be specified as 1, and the control parameter NREAL set to zero. If the dimensional drawdown and time are to be printed, the basic values of these parameters in proper units must be read in and NREAL set to 1. Although the printout will not show the actual units of time and drawdown as well as other basic parameters, the user should know what units they are. The range of dimensionless times to be used depends on the basic values of permeability, specific storage and length as indicated by Eq. 21.

1.6 Basis for Selection of Method

There are two major numerical methods for analyzing groundwater flow, viz. the finite difference method and the finite element method. The latter method was used in this program because it is more versatile and can match irregular boundaries more closely. Furthermore, its coefficient matrix is always symmetric, positive and definite, so the large number of simultaneous equations can be solved effectively.

The program utilizes basic tetrahedra as the building block for finite elements, instead of other more sophisticated higher order elements. The advantage of using tetrahedral elements lies in the fact that their element stiffness matrix can be expressed in simple forms, as shown in Eq. 10, thus no numerical integration is needed.

1.7 Accuracy, Limitations, and Restrictions

As presented in Chapter III, the solutions obtained from this program were compared with the exact solution by Theis and the finite element solution by cylindrical elements, and they were found in good agreements. The accuracy of the solution depends on the size and shape of the finite elements employed. Experience has shown that if the length ratio of triangles or quadrilaterals on the plan view is not greater than 2 or 3 and that on the side view not greater than 4 or 5, satisfactory results can usually be obtained.

The limitations of this program can be best judged from the assumptions by which the program was developed. These assumptions are discussed in Chapter II. The assumption that the initial piezometric surface is a plane greatly simplifies the program by eliminating the actual piezometric head as an input parameter. If the initial piezometric surface is not a plane and the initial piezometric head at each nodal point is specified, the program can be easily modified by using the piezometric head, instead of the drawdown, as a dependent variable. The assumptions that the top and bottom boundaries of the aquifer are planes and that the circumferential boundaries are vertical not only eliminate the tedious work of assigning nodal coordinates but also facilitate the visualization and graphical presentation of the aquifer. These limitations can be easily eliminated by the user if he can take out the subroutine PLANE and assign the coordinates for each of the nodal points. The assumption that the three principal directions of permeability are the same for all elements and one of them is along the vertical, or z axis, is to simplify the practical application of the program. If the principal directions are not the same, the user can modify the program by specifying the principal directions for each element, computing the stiffness matrix with respect to the principal directions, and then transforming it to global coordinates. All these can be included in a subroutine without changing the original program.

It is believed that the assumptions employed in the development of this program are quite realistic and should not limit severely the applicability of the program. There is practically no restriction on the size of aquifer or the number of elements to be used. If the number of equations to be solved is large, the equations should be divided into a sufficient number of partitions, so that the storage required does not exceed the capacity of the computer.

1.8 Functional Information

The computer program is formed by the following parts: (1) a main program; (2) nine subprograms; viz. NODP, PLANE, BAPTI, STIFI, FEM, TETRA, FASEI, PHPLOT, and SESOL; (3) six subroutines, viz. PLOTS, PLOT, SCALE, LINE, NUMBER, and SYMBOL, developed by the Calcomp Company specifically for the Calcomp plotter; and (4) a subroutine called TIMER taken from the subprogram library of the University of Kentucky Computing Center. Because the subroutines in (3) and (4) are machine dependent and some of them are written in assembler language, they are not listed in the source program. The user should check the type of computer available and replace these subroutines with the ones designed specifically for his computer.

1.8a Main Program

The main program takes care of all the housekeeping work for the computation. It controls the flow of the analysis as follows: (1) read in all data input and print them out; (2) determine the size of array for the variables used in the subprograms; (3) call subprograms NODP, PLANE, BAPTI, STIFI, and PHPLOT; (4) print out drawdown at specified nodal points; and (5) plot the plan view and the average drawdown at a given time for the layer plane assigned.

A simplified flow chart is shown in Figure A2.

1.8b Subprogram NODP

This subprogram is used to (1) determine the element number with permeabilities different from those of the standard element or storage coefficient different from 1; (2) compute the total number of nodal points in each element; and (3) relate the nodal number in each element to the overall nodal number.

A simplified flow chart is shown in Figure A3.

1.8c Subprogram PLANE

This subprogram is used to (1) determine the coefficients of plane equations for the top and bottom boundaries of the aquifer; and (2) determine the coordinates for each node.

A simplified flow chart is shown in Figure A4.

1.8d Subprogram BAPTI

This subprogram is used to (1) calculate total number of equations; (2) determine number of equations in each partition; (3) compute the maximum half band width; and (4) number the starting and ending elements for each partition so that the appropriate stiffness matrix can be read from disk no. 8.

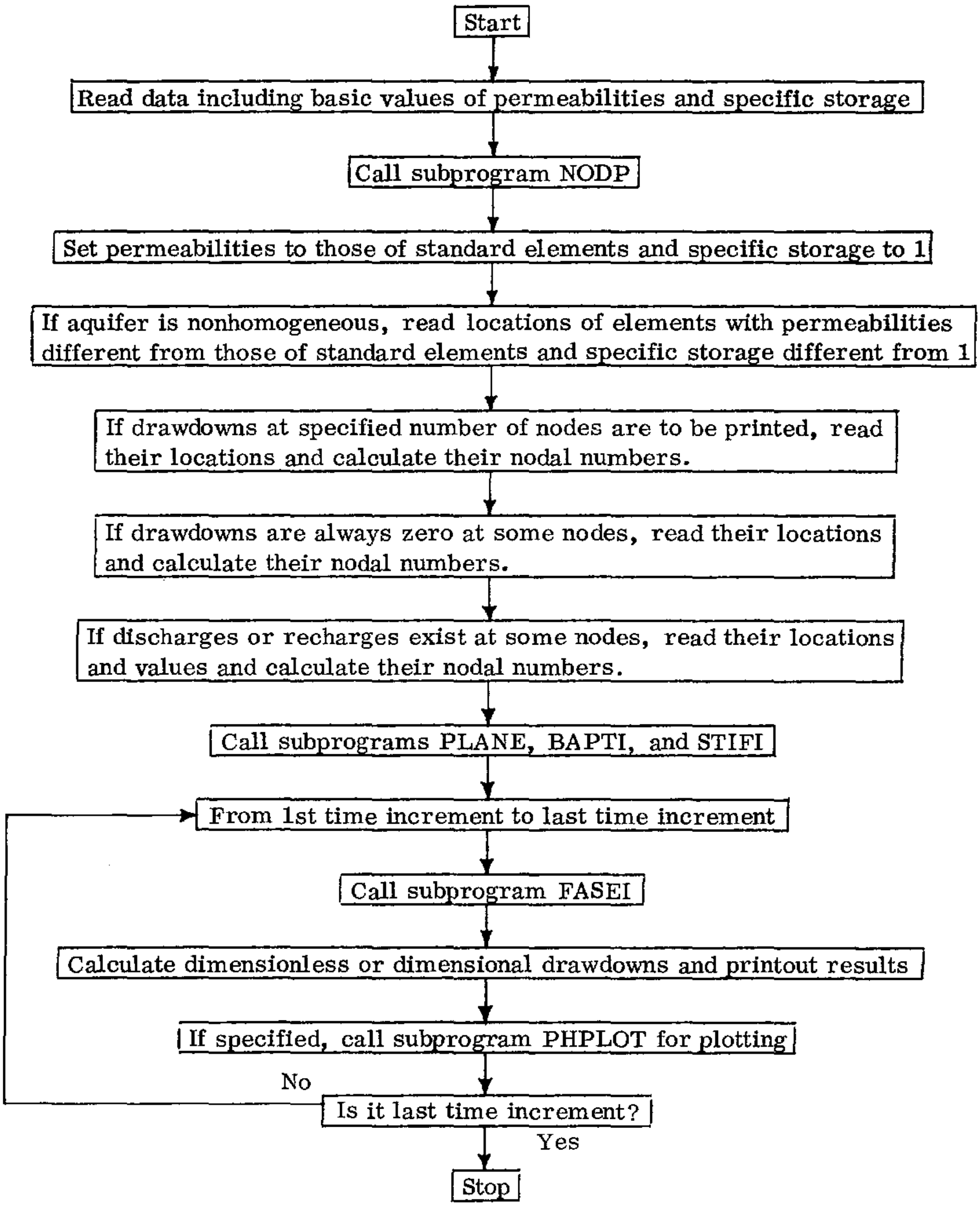


Figure A2. Flow chart for main program.

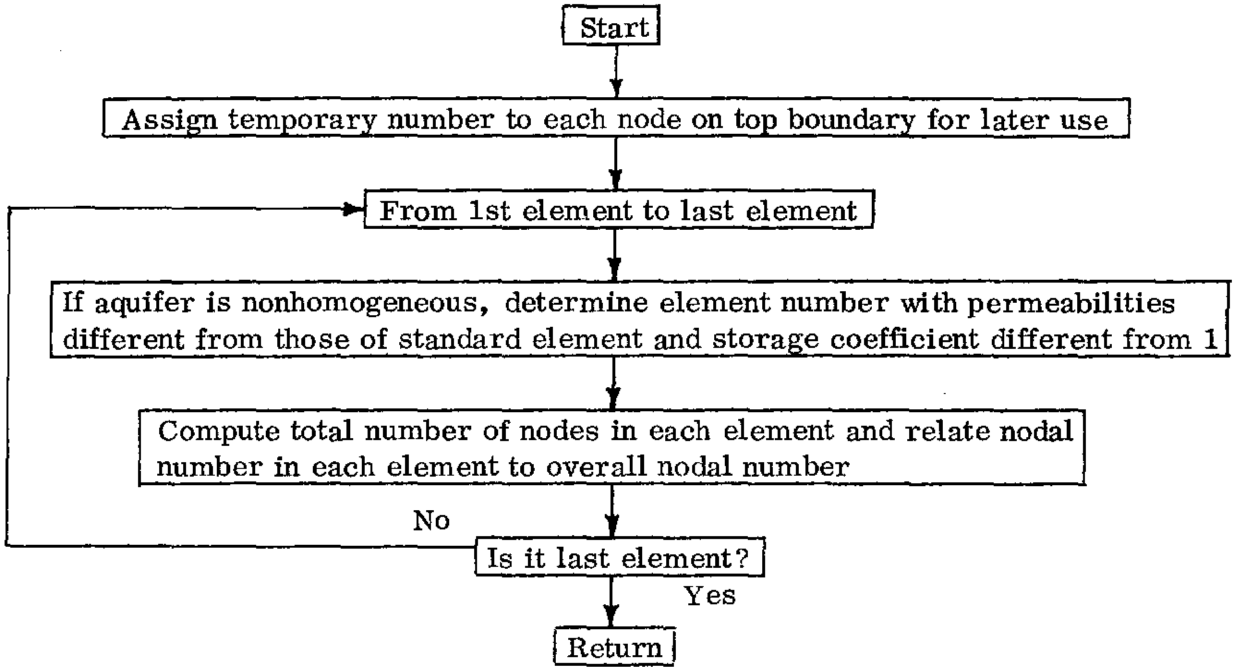


Figure A3. Flow chart for subprogram NODP.

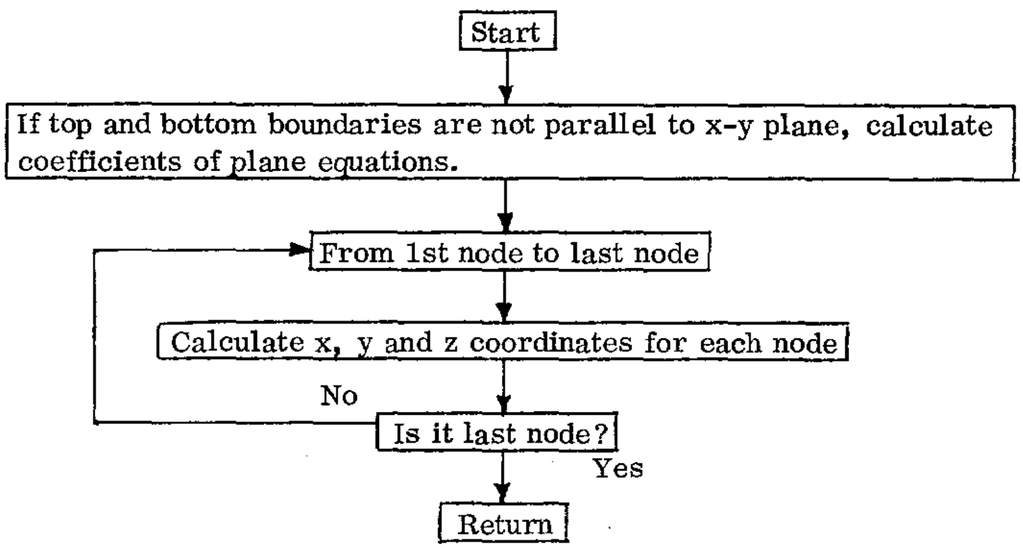


Figure A4. Flow chart for subprogram PLANE.

A simplified flow chart is shown in Figure A5.

1.8e Subprogram STIFI

This subprogram is used to (1) check whether the elements are physically the same so that only the stiffness matrix of the first element needs be calculated; (2) call subprogram FEM to calculate the stiffness matrix of each six- or eight-node element; and (3) store the stiffness matrix of each element in disk no. 8.

A simplified flow chart is shown in Figure A6.

1.8f Subprogram FEM

This subprogram is used to (1) transform the tetrahedral nodal number into the nodal number of the six- or eight-node element; (2) call subprogram TETRA to obtain the stiffness matrices of each tetrahedron; and (3) superimpose the stiffness matrices of tetrahedra to form those of the six- or eight-node element.

A simplified flow chart is shown in Figure A7.

1.8g Subprogram TETRA

This subprogram is used to compute the stiffness matrices [H] and [P] of a tetrahedral element.

A simplified flow chart is shown in Figure A8.

1.8h Subprogram FASEI

This subprogram is used to (1) form [D] matrix for determining the column matrix on the right side of simultaneous equations; (2) form [C] matrix and store it together with the column matrix in the disk; and (3) call subprogram SESOL for solving the simultaneous equations for drawdown.

A simplified flow chart is shown in Figure A9.

1.8i Subprogram PHPLOT

This subprogram plots a plan view and prints out the average drawdown of each element on a given layer plane. The drawdown is printed directly on the corresponding element to represent the value at the centroid which is marked by a cross. If the element is too small to accommodate the printing of drawdown,

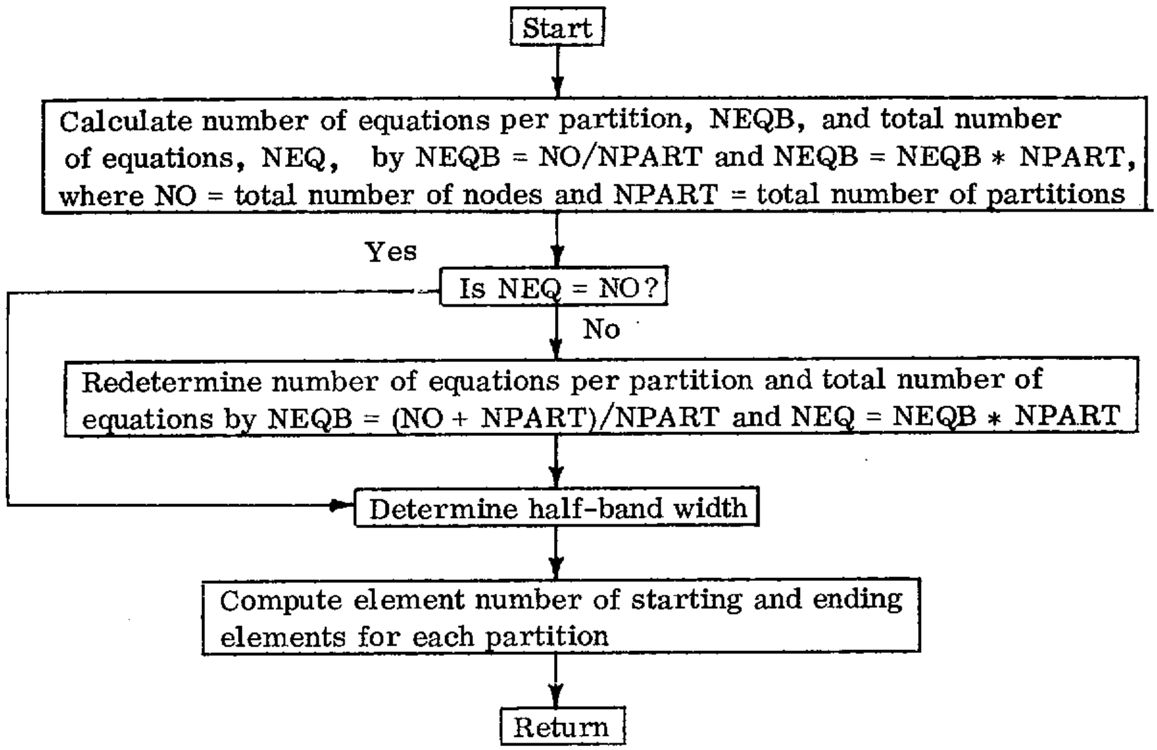


Figure A5. Flow chart for subprogram BAPTI.

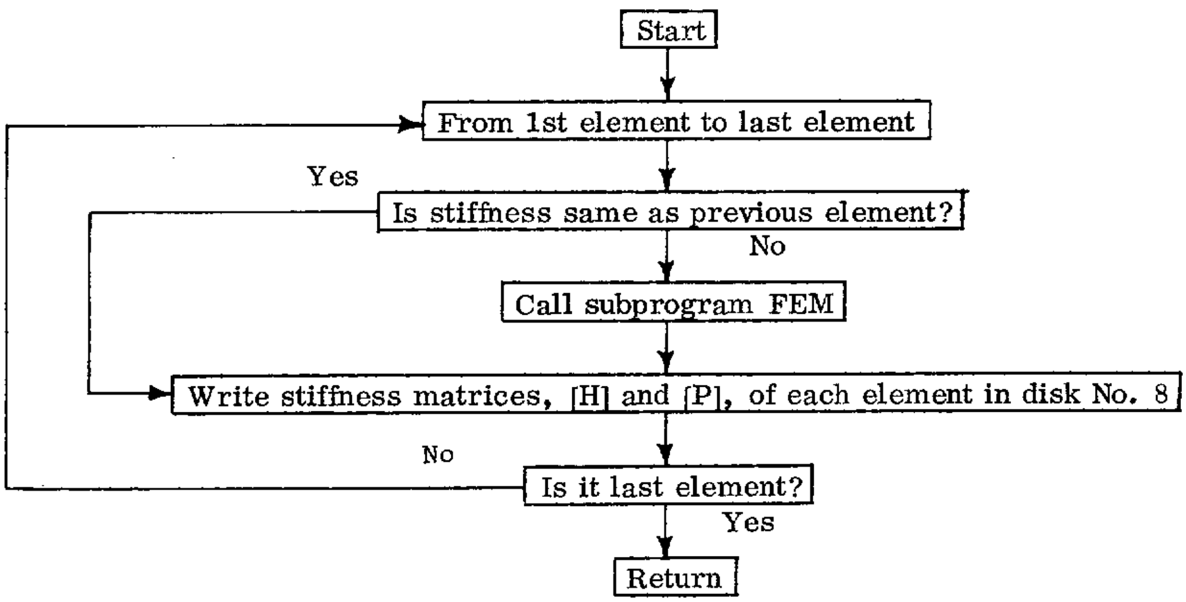


Figure A6. Flow chart for subprogram STIFI.

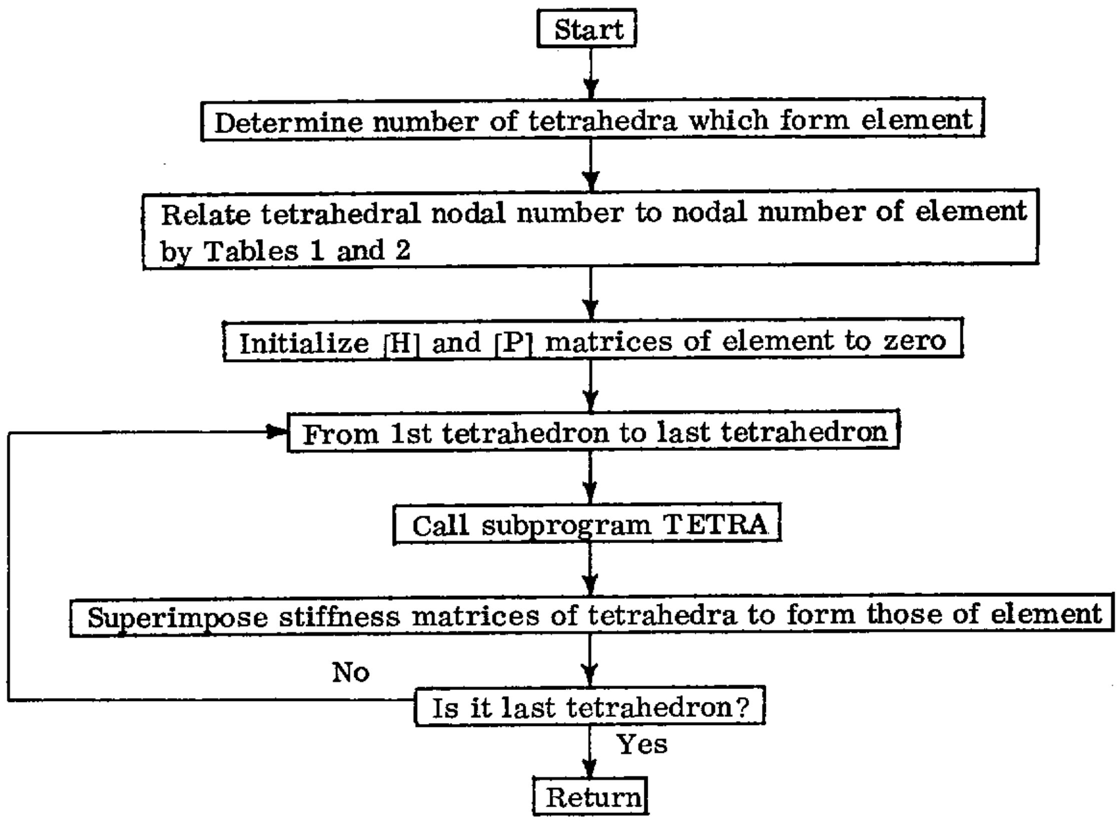


Figure A7. Flow chart for subprogram FEM.

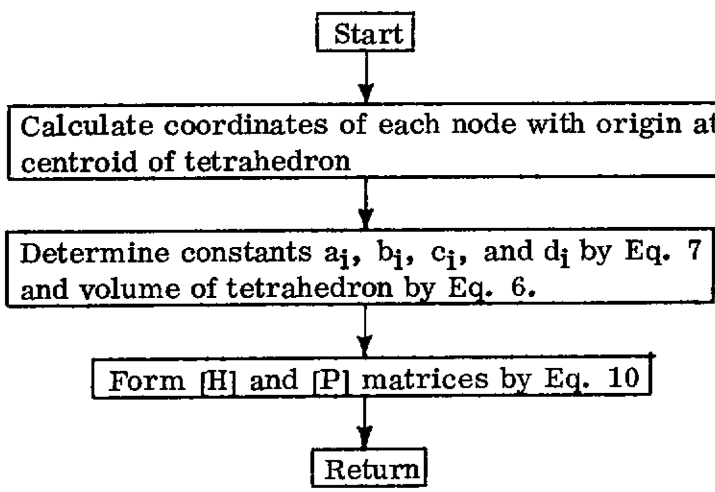


Figure A8. Flow chart for subprogram TETRA.

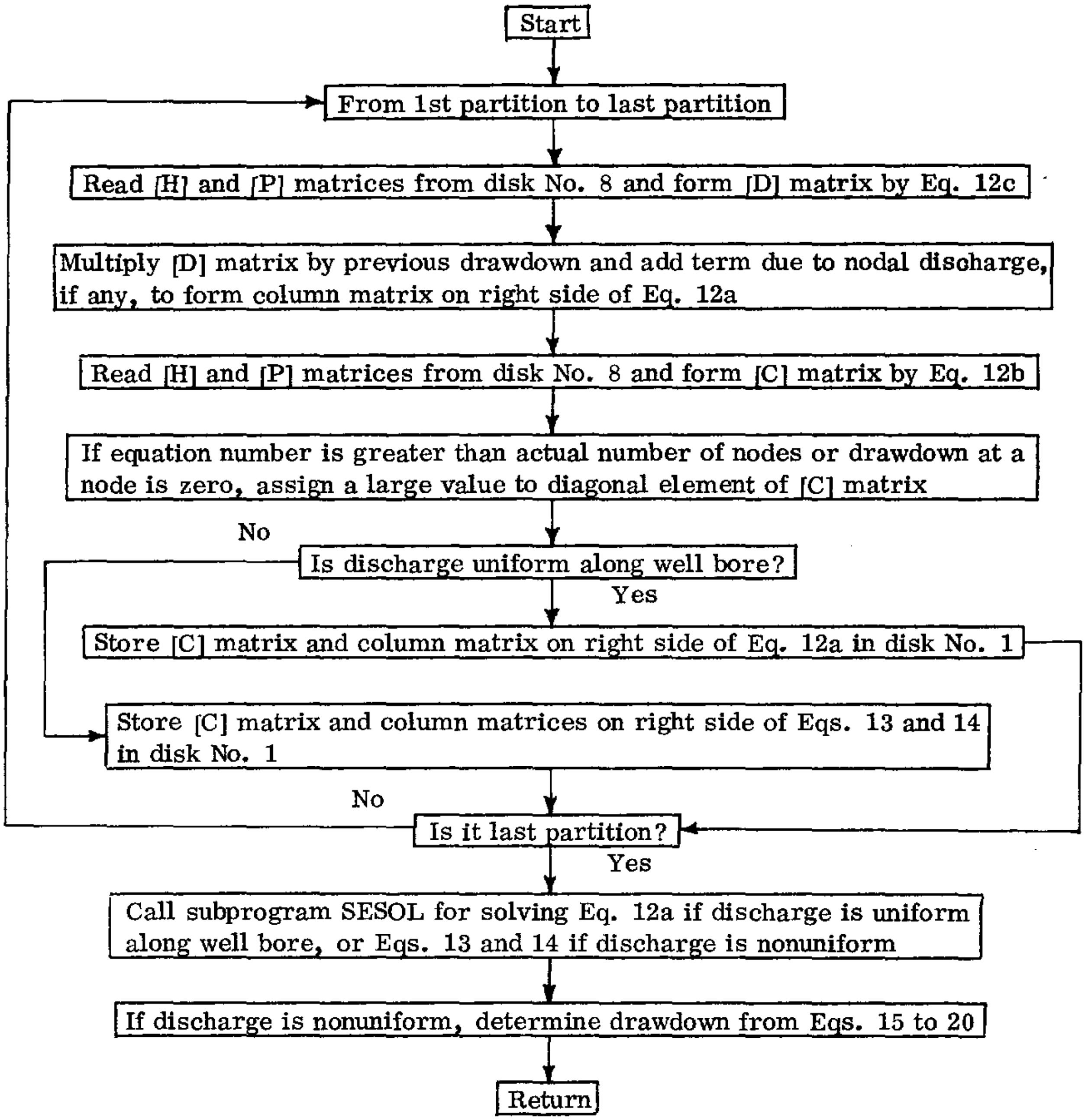


Figure A9. Flow chart for subprogram FASEI.

an alphabetical character will be printed inside the element, and the actual value of drawdown will be printed outside the plan view. If the element is so small that even an alphabetical character cannot be accommodated, a cross will be marked on the first element and the drawdowns will be printed outside in prescribed sequence starting from the first element.

A simplified flow chart is shown in Figure A10.

1.8j Subprogram SESOL

This subprogram is used to solve the linear simultaneous equations by partitions, and the results will be the drawdown of each nodal point at the end of each pumping period. This subprogram was described in detail by Wilson, Bathe and Doherty [1974] and will not be reported here.

1.8k Plotting Subroutines

These subroutines are not listed in the source program. Those who are interested in these subroutines are suggested to consult "Users' Manual for the Calcomp Plotter with OS/360" published by the University of Kentucky Computing Center or other similar publications by the Calcomp company.

(1) Subroutine PLOTS

This subroutine is used to initialize plotting.

(2) Subroutine PLOT

This subroutine determines a new position or new set of x, y coordinates for the pen.

(3) Subroutine SCALE

This subroutine determines the scaling factor which is the value of length corresponding to one inch of plotting paper.

(4) Subroutine LINE

This subroutine produces a line plot of the pairs of values in x and y arrays. It computes the coordinates of each plotted point according to the data values in each array and the respective scale factors.

(5) Subroutine NUMBER

This subroutine produces a plot for an integer or a real number.

(6) Subroutine SYMBOL

This subroutine produces plot annotation at any angle and in

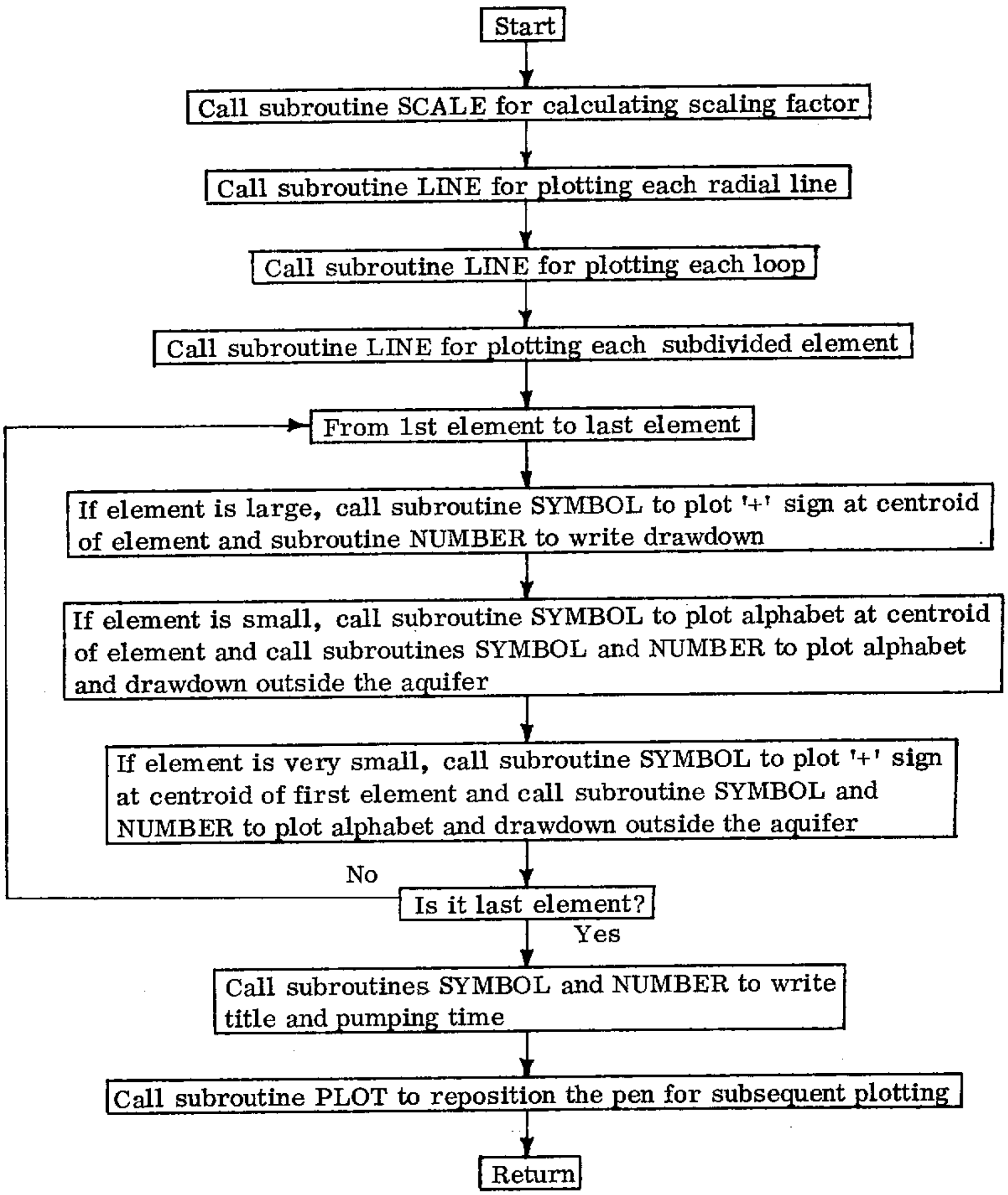


Figure A10. Flow chart for subprogram PHPLOT.

practically any size. The symbols can be titles or symbols from the symbol table.

1.81 Subroutine TIMER

This subroutine is used to count the time in miliseconds for solving simultaneous equations and the accumulated time from the beginning. It is an assembler language subroutine which can be called by a Fortran program to determine elapsed times during execution of a program. The subroutine is called by a statement of the form.

CALL TIMER(J,K)

in which J = 0,1, or 2 and represents the option desired; and K = an integer *4 variable for elapsed time and may be omitted for option 0. The three options are explained below.

Option 0 - sets the timer to zero, overriding any previous timer that may have been set.

Option 1 - returns in K the time in miliseconds since the timer was last set, but does not reset the timer.

Option 2 - returns in K the elapsed time since the timer was last set, and then resets the timer to zero.

1.9 Definition of Technical Terms

Anisotropic aquifer - An aquifer the permeability of which depends on the direction of flow.

Artesian well - A well tapping a confined aquifer in which the water level rises above the top of the aquifer.

Basic values - The values of permeability, specific storage, discharge, length, transmissibility, and storage coefficient used to determine the dimensionless drawdown and the dimensionless time.

[C] matrix - An overall stiffness matrix, or the coefficient matrix of simultaneous equations.

Confined aquifer - An aquifer bounded above and below by impermeable bed or beds of distinctly lower permeability than that of the aquifer itself.

[D] matrix - A matrix used to determine the column matrix on the right-hand side of simultaneous equation.

Discharge - Rate of flow from an aquifer at a given node in terms of volume per unit time.

Drawdown - The lowering of the piezometric surface caused by pumping.

Drawdown factor - Dimensionless drawdown expressed as (actual drawdown x basic transmissibility)/basic discharge.

Element number - A number assigned to each finite element.

Element stiffness matrix - The stiffness matrix of a tetrahedron as given by Eq. 10.

Finite element - A triangular prism or hexahedron formed by tetrahedra.

Finite well - A well with a finite radius.

Full aquifer - An aquifer which is not symmetric with respect to the z axis so the whole aquifer is used in the finite element analysis.

Full radial line - A radial line which originates from the center of well.

Fully penetrating well - A well penetrating the entire thickness of aquifer.

Global coordinates - x, y and z coordinates with the z coordinate coinciding with the center of well.

Homogeneous aquifer - An aquifer having the same permeabilities and specific storage everywhere.

Infinitesimal well - A well with a radius approaching zero.

Isotropic aquifer - An aquifer having the same permeability in every direction.

Layer planes - Planes which divide an aquifer into layers of finite elements.

Layer plane number - A number assigned to each layer plane with the top boundary of aquifer designated as layer plane no. 1.

Loop - A series of straight lines which connect corresponding nodes on two adjoining radial lines.

Loop number - A number assigned to each loop with the center of well designated as loop no. 1.

Node or nodal point - A point in the aquifer, which is the corner of one or more finite elements.

Nodal number - A number assigned to each node.

Nonhomogeneous or heterogeneous aquifer - An aquifer composed of two or more zones, each having a different set of permeabilities or specific storage.

Nonuniform discharge - A method of analysis in which the discharge along with well bore is not assumed uniformly distributed.

Partial aquifer - An aquifer which is symmetric with respect to the z axis so only a slice of the aquifer is used in the finite element analysis.

Partial radial line - A radial line which starts from a specified loop other than loop 1, which is the center of well.

Partially penetrating well - A well penetrating only partial thickness of aquifer.

Partition - A method of solving simultaneous equations in which the equations are divided into a number of partitions.

Permeability - A property of aquifer defined as the flow of water through a unit cross-sectional area normal to the direction of flow when the hydraulic gradient is unity.

Piezometric surface - The surface to which the water from a given aquifer will rise under its full head.

Previous drawdown - The drawdown at the end of previous time increment.

Radial line number - A number assigned to each radial line including all full and partial radial lines.

Recharge - Rate of flow to an aquifer at a given node in terms of volume per unit time.

Special element - An element which may exist in two different forms and thus requires specification by the user as to the form employed.

Specific storage - A property of aquifer defined as the amount of water in storage released from a unit volume of aquifer under a unit decline of head.

Standard element - An element having prescribed values of permeabilities in three principal directions.

Storage coefficient - The volume of water released from storage per unit of surface area of the aquifer per unit change in head, which is also equal to the product of specific storage and the thickness of aquifer.

Subdivision - A process in which the aquifer is divided by a series of full and partial radial lines.

Subdivided left element - The triangular prism on the left side of a subdivided mid element.

Subdivided mid element - A triangular prism the plan view of which is formed by a node on the partial radial line and two nodes, one on each adjacent radial line.

Subdivided right element - The triangular prism on the right side of a subdivided mid element.

Thickness of aquifer - The distance between top and bottom boundaries of aquifer along a vertical line which, in the case of aquifer with variable thickness, is measured along the center of well, or z axis.

Time - The time since pumping started.

Time factor - Dimensionless time expressed as (actual time x basic transmissibility)/(storage coefficient x square of basic length).

Transmissibility - The rate at which water will flow through a unit width of aquifer under a unit hydraulic gradient, which is also equal to the product of permeability in the radial direction and the thickness of aquifer.

Uniform discharge - A method of analysis for infinitesimal well in which the discharge along the well is assumed uniformly distributed.

PART II USAGE INFORMATION

2.1 Programming Language, Equipment and Operating System

The computer program was written in Fortran IV for an IBM 360 computer, model 65, which is used presently by the University of Kentucky. The operating system is RELEASE 21.6. The control cards presented in section 2.4 are based on Fortran G compiler with Calcomp plotter. The program executes in 268 K main core and does not require any external tapes for the intermediate storage of data, other than disks 1, 2, 3, 4, 5, 6 and 8 provided by the computer.

As mentioned before, the computer program consists of one main program, nine subprograms, six Calcomp subroutines and one IBM subroutine. The source decks for main program and subprograms are complete in themselves and do not depend on any outside subroutines. However, the six Calcomp subroutines for plotting and the IBM subroutine TIMER are not included in the source decks. Unless the same equipment is available, the user has to supply his own subroutines or use the subroutines written specifically for his equipment.

The program can be executed without a plotter by the following steps: (1) change the job control card, "// EXEC FORTGCLP" to "// EXEC FORTGCLG", (2) take out the three statements for calling PLOTS and PLOT at the very beginning of the main program, (3) take out the two statements for calling PLOT and EXIT at the end of the main program, and (4) take out subroutine PHPLOT. The program can be executed without a timer simply by taking out the three statements for calling TIMER and the four cards following the last statement for calling TIMER.

2.2 Input Requirements

All the input data required for this program are located in the main program. Before preparing the input data the user must have a general idea on the dimension of the aquifer, the location of the well, and the points at which discharge, recharge or zero drawdown occurs, so that he can tentatively divide the aquifer into finite elements. The major requirements for preparing input data are described as follows:

2.2a Division of Aquifer into Finite Elements

Although the aquifer is divided into a set of six- and eight-node elements, the basic components to form these elements are only three, viz. radial lines, loops, and layer planes. The radial lines may be a full line radiated from the center of well or a partial radial line starting at a given loop. They must be arranged in a regular pattern, so that the nodal and element numbers can be generated automatically. Starting from the positive x axis in a counterclockwise direction, each radial line is assigned a number. Each loop is also assigned a number, counting the center of the well as loop no. 1 and proceeding outwards. Unless a loop intersects the boundary of aquifer, it must be continuous and form a closed figure. The layer planes are numbered from top to bottom.

Figure A11 shows how an aquifer is divided into finite elements. The numbers marked at each node are the nodal numbers, and those inside the parenthesis are the element numbers. To divide the aquifer according to Figure A11, the following input data are needed:

(1) Total number of radial lines

There are 12 radial lines, each marked by a number inside the circle. A partial radial line, which starts at loop 4, always exists between two full radial lines. The angle between each radial line and the first radial line must be specified.

(2) Total number of loops

There are five loops including the one at the center of well. The loops are drawn in heavier lines, and their numbers are marked.

(3) Total number of layer planes

There are three layer planes. The z coordinate of each layer plane at the center of well, or along the z axis, must also be specified.

(4) Total number of nodes on each radial line

The number of nodes on each radial line may be equal to or smaller than the total number of loops and is tabulated below.

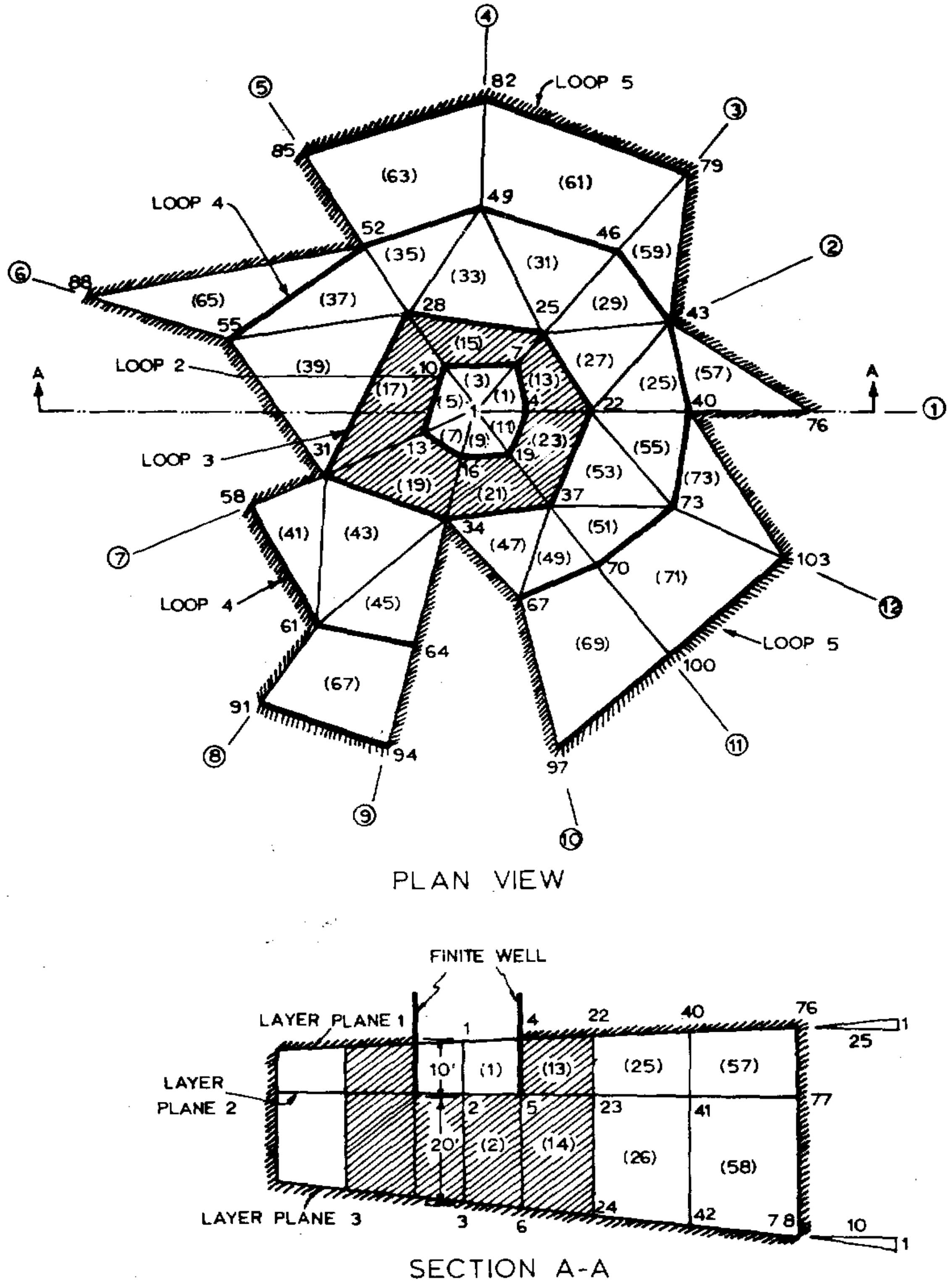


Figure A11. Division of aquifer and numbering system.

Radial line no.	1	2	3	4	5	6	7	8	9	10	11	12
Number of nodes	5	4	5	5	5	5	4	5	5	5	5	5

The number of nodes on each radial line is actually the number of loops intersecting each radial line. In counting the number of intersections, the partial radial line is considered as a full line extending to the well.

(5) Radial distances from the center of well to each loop measured along each radial line.

At the center of well, the radial distance is zero. The radial distance to a loop along a partial radial line is considered zero if the partial line does not actually intersect the loop. For example, the radial distances for nodes on radial line 1 are 0.0, 10.0, 22.0, 40.0, and 62.0 ft ; while those on radial line 2 are 0.0, 0.0, 0.0, and 40.0 ft.

(6) Total number of elements per layer between two radial lines, or on the left of each radial line

For a full aquifer, it is necessary to read in the number of elements on the left of the last radial line; whereas for a partial aquifer, no such reading is needed. If an aquifer is divided by both full and partial radial lines, the subdivided mid elements should not be counted. The subdivided mid element is an element formed by a node at which the partial radial line starts and two nodes on the previous loop, as shown by elements 27, 33, 39, 43, 47 and 53. The element on the left of a subdivided mid element is defined as a subdivided left element, such as element 29, while that on the right is defined as a subdivided right element, such as element 25. In counting the number of elements between two radial lines, all partial radial lines must be extended to the well, and all elements counted except the subdivided mid elements. The number of elements on the left of each radial line is listed below.

Radial line no.	1	2	3	4	5	6	7	8	9	10	11	12
Number of elements	4	4	4	4	4	2	3	4	2	4	4	4

2.2b Location of Nodal Points

If the drawdown at only a specified number of nodes is to be printed, it is not necessary for the user to determine their nodal numbers. Instead, three numbers; viz. a radial line number, a loop number, and a layer plane number, are used to locate a nodal point. For nodal points along the center of well, the radial line number is assigned as 1. For example, nodes 2, 22, 51, and 103 in Figure A11 can be located as follows:

Nodal no.	Radial line no.	Loop no.	Layer plane no.
2	1	1	2
22	1	3	1
51	4	4	3
103	12	5	1

When the three numbers are given, the computer will determine the actual nodal number and print it out together with the three location numbers and the resulting drawdown.

2.2c Location of Elements

In much the same way as nodal points, the location of an element is also by three numbers: (1) the radial line number on the right side of the element, (2) the loop number at the back of the element, and (3) the layer plane number on the top of the element. The subdivided mid element and the subdivided left element have the same location numbers, and their radial line number is the number of the partial radial line. For example, elements 1, 14, 42, 43, 45, 65 and 73 can be located as follows:

Element no.	Radial line no.	Loop no.	Layer plane no.
1	1	2	1
14	1	3	2
42	7	4	2
43	8	4	1
45	8	4	1
65	5	5	1
73	12	5	1

Elements 39, 40, 65, 66, 73 and 74 are called special elements because they can exist in two different forms. Elements 73 and 74, which have two nodes on the ith radial line and one node on the (i + 1)th radial line, are called special elements of type 1. If they are not designated as special elements, the computer will not recognize them as triangular prisms but as hexahedra by moving boundary 40-103 to 76-103. Elements 65 and 66, which have one node on the ith radial line and two nodes on the (i + 1)th radial line, are called special elements of type 2. If they are not designated as such, the computer will consider them as hexahedra by moving boundary 52-88 to 85-88. Elements 39 and 40, which are subdivided mid elements without a subdivided left element belong to special element of type 3. If they are not designated as such, the computer will assume that a subdivided left element exists by moving boundary 31-55 to 55-58. Note that elements 57 and 59 are not considered as special elements because they are always triangular prisms and there is no way for them to become hexahedra.

2.3 Description of Input Data

The input data and their array sizes, if any, are listed in section 2.7. This section will describe these variables in more detail together with the format to be used. Variables will be described in the order as they appear in the program. Examples of preparing input data are presented in section 2.8.

2.3a General Formats

The input data are punched in data cards from column 1 to column 80. Only five formats are used.

(1) Fixed mode format (type I): (16I5)

This format indicates that each integer is punched within five columns and the maximum number of integer data to be punched in one card is 16.

(2) Floating mode format (type II): (8F10.5)

This format indicates that a maximum number of eight floating mode values can be punched in one card and each occupies ten columns with five columns for decimal points. This format is used for all real value variables except for the pumping times.

(3) Floating mode format (type III): (6F13.5)

This format is used in section 2.3g for pumping times only. Six floating mode values can be punched in one card, and each occupies 13 columns with five columns for decimal points.

(4) Floating mode format (type IV): (2E10.3, 5F10.3)

This format is used only in section 2.3c for basic values and permeabilities of the standard element. The first two entries are in E format and the remaining in F format.

(5) Mixed mode format (type V): (3I5, F10.5)

This format is used only in section 2.3n when discharge or recharge occurs at some nodes, or when KD $\neq$ 0. Three integers are punched in the first fifteen columns, each occupying five columns, followed by a real number occupying 10 columns.

2.3b General Controlling Parameters

(1) Variables: NP, NR, NZ, NPART, NCOLN, NT, NFG, NM, NFW, JD, LOP, NDP, KP, KS, NTB, NB, IE, NPA, MD, NREAL, KD, KH, MPT.

(2) Format: Use type I with the first 16 variables in one card and the remaining 7 in a second card.

(3) Description:

NP is the maximum number of nodes in a radial line. Each radial line may have a different number of nodes, and NP is the maximum number among all radial lines.

NR is the total number of radial lines, including all full and partial radial lines.

NZ is the total number of layer planes, which is equal to the number of finite element layers plus one.

NPART is the total number of partitions used in solving simultaneously equations. The selection of the number of partitions depends on the storage capacity of the computer. In principle, the least number of partitions should be used in commensurate with the computer storage.

NCOLN is a parameter for assuming the type of well discharge. NCOLN = 1 if the discharge is assumed uniform along the well bore. Otherwise, NCOLN = 2. The latter case should be used for a partially penetrating finite well.

NT is the total number of times at which drawdowns are computed, which is equal to the total number of time increments plus one.

NFG is a parameter indicating whether the aquifer is full or partial. NFG = 0 for full aquifers and 1 for partial aquifers. A full aquifer has at least one closed loop, but a partial aquifer does not have any loop which is closed. Partial aquifers are used for axisymmetric problems in which only a slice of the aquifer is taken for analysis.

NM is a parameter indicating whether the aquifer is homogeneous or nonhomogeneous. NM = 0 for homogeneous aquifer, and NM = 1 for nonhomogeneous aquifer.

NFW is a parameter indicating whether the well is infinitesimal or finite in radius. NFW = 0 for infinitesimal well, and NFW = 1 for finite well. The circumference of a finite well is defined by loop 2, while an infinitesimal well is located at the origin, or loop 1.

JD is the total number of nodes at which drawdowns are to be printed. JD = 0 if the drawdowns at all nodes are to be printed.

LOP is a parameter indicating the orientation of top and bottom boundaries of the aquifer. LOP = 1 if top and bottom boundaries are horizontal, or parallel to x-y plane. LOP = 2 if they are parallel to each other but not to x-y plane. LOP = 3 if both are arbitrary.

NDP is the total number of layer planes penetrated or tapped by the well.

KP is the total number of elements the permeabilities of which are different from those of the standard element. KP = 0 for homogeneous aquifers because every element has the same permeabilities and is considered as a standard element.

KS is the total number of elements the specific storage of which is different from the basic value. KS = 0 for homogeneous aquifers. For nonhomogeneous aquifers, the specific storage which occurs most frequently will be taken as the basic value, and KS will be the number of elements having a specific storage different from the basic value.

NTB is a parameter indicating the type of boundary. NTB = 0 if the number of nodes in all radial lines is the same, or every loop forms a closed figure. Otherwise, NTB = 1.

NB is the total number of subdivisions. A subdivision occurs when a new set of partial radial lines starts at a given loop. NB = 0 if there is no subdivision, or no partial radial lines.

IE is the total number of special elements. Definition of special elements is presented in section 2.2c. IE = 0 if there is no special element.

NPA is the number of partitions actually employed in the first time increment. If NPA = NPART, all the partitions are used in solving the simultaneously equations. To save the computer time for large jobs, NPA may be set to 1 or 2, so only the first one or two partitions be used in the first time increment.

MD is the total number of layer planes to be plotted. MD = 0 if no plotting is contemplated. Maximum value of MD should not be greater than NZ.

NREAL is a parameter indicating whether the drawdown and time are printed in real or dimensionless values. NREAL = 0 for dimensionless values, and NREAL = 1 for real values.

KD is the total number of nodes at which constant discharge or recharge is specified. KD = 0 if no discharge or recharge, other than the discharge at the well, takes place.

KH is the total number of nodes at which drawdowns are maintained zero.

MPT is the total number of pumping times at which the aquifer and the drawdown are to be plotted. MPT = 0 if no plotting is contemplated. Maximum value of MPT should not be greater than NT.

2.3c Basic Values and Permeabilities of Standard Element

(1) Variables: BK, BS, BQ, BL, SK(1), SK(2), SK(3).

(2) Format: Use type IV.

(3) Description:

BK is the basic value of permeability. Although any permeability may be used as a basic value, the general rule is to take the permeability which occurs repeatedly in various elements as the basic value. All other permabilities are read in as a ratio to the basic value.

BS is the basic value of specific storage, or the specific storage which occurs repeatedly in various elements. All other values of specific storage are read in as a ratio to the basic value.

BQ is the basic value of discharge, which is taken as the total discharge from the well per unit time. All discharges and recharges are read in as a ratio to the basic value. BQ is always the total discharge from the well no matter whether full or partial aquifers are involved.

BL is the basic value of length. All lengths, distances, and coordinates are read in as a ratio to the basic length.

SK(1), SK(2), and SK(3) are respectively the permeabilities in x, y and z directions of a standard element. These permeabilities are read in as a ratio to the basic value. For isotropic aquifers, SK(1) = SK(2) = SK(3) = 1. For anisotropic aquifers, one or two of these variables will have a value of 1.

2.3d Location of Nodes for Drawdown

(1) Variables: NNP(I, 1), NNR(I, 1), NNZ(I, 1); where I is from 1 to JD.

(2) Format: Use type I. Only three values are punched in one card, each occupying five columns, so only the first 15 columns are used in one card. A total of JD cards is required. No data card is needed if JD = 0.

(3) Description:

NNP(I, 1) is the loop number, NNR(I, 1) is the radial line number, and NNZ(I, 1) is the layer plane number. The use of these three numbers to locate a nodal point is described in section 2.2b.

2.3e Starting Loop for Partial Radial Lines

(1) Variable: NS(I); where I is from 1 to NB.

(2) Format: Use type I. The number of values punched in one card depends on the number of subdivisions. Each value occupies five columns. If NB = 0, no data card is needed.

(3) Description:

NS(I) is the loop number from which the partial radial lines for the ith subdivision start.

2.3f Number of Elements Per Layer Between Two Radial Lines

(1) Variable: NND(I); where I is from 1 to NR.

(2) Format: Use type I. Maximum number of entries is 16 per card, each occupying five columns.

(3) Description:

NND(I) is the total number of elements per layer between two radial lines, or on the left of the ith radial line. The method of counting is described in section 2.2a (6).

2.3g Pumping Times

(1) Variable: T(I); where I is from 1 to NT.

(2) Format: Use type III. Only six values of T(I) are punched in one card, each occupying 13 columns. The first entry in the first card will always be 0.00000, which is the value for T(1).

(3) Description:

T(I) is the time since pumping started. T(I) is read in as a dimensionless time in logarithmic scale.

2.3h Maximum Radial Distance from Well to Aquifer Boundary

(1) Variable: RR

(2) Format: Use type II. This value is punched in the first ten columns of the data card.

(3) Description:

RR is the maximum radial distance from the center of well to the boundary of aquifer. This information is used to determine the scale for plotting. RR may be equal to or greater than the maximum length of radial lines actually used.

2.3i Pumping Time for Plotting

(1) Variable: TP(I), where I is from 1 to MPT.

(2) Format: Use type III.

(3) Description:

TP(I) is the pumping time at which drawdowns are plotted.

2.3j Layer Plane Number for Plotting

(1) Variable: ME(I); where I is from 1 to MD

(2) Format: Use type I. If MD = 0, this part of input is not needed.

(3) Description:

ME(I) is the ith layer plane on which a plan view of the aquifer and the drawdown are to be plotted.

2.3k Location of Elements with Permeabilities Different from Standard Element

(1) Variables: NNP(I, 2), NNR(I, 2), NNZ(I, 2); where I is from 1 to KP.

(2) Format: Use type I. No data card is needed if NM = 0 or KP = 0.

(3) Description: Refer to sections 2.2c and 2.3c for details.

2.3l Location of Elements with Specific Storage Different from Basic Value

(1) Variables: NNP(I, 3), NNR(I, 3), NNZ(I, 3); where I is from 1 to KS.

(2) Format: Use type I. No data card is needed if NM = 0 or KS = 0.

(3) Description: Refer to sections 2.2c and 2.3c for details.

2.3m Location and Type of Special Elements

(1) Variables: NNP(I, 4), NNR(I, 4), NNZ(I, 4), IS(I); where I is from 1 to IE.

(2) Format: Use type I. Four entries in each card. No data is needed if IE = 0.

(3) Description: Refer to sections 2.2c for details.

2.3n Location of Nodes with Given Discharge or Recharge

(1) Variables: NNP(I, 5), NNR(I, 5), NNZ(I, 5), FQ(I); where I is from 1 to KD.

(2) Format: Use type V. No data card is needed if KD = 0.

(3) Description: Refer to section 2.2b for details. FQ(I) is the discharge (positive) or recharge (negative) as a ratio to the basic value.

2.3p Location of Nodes with Zero Drawdown

(1) Variable: NNP(I, 6), NNR(I, 6), and NNZ(I, 6); where I is from 1 to KH.

(2) Format: Use type I. No data card is needed if KH = 0.

(3) Description: Refer to section 2.2b for details.

2.3q Orientation of Top and Bottom Boundaries

(1) Variable: ZC(K, I, J); where K is from 1 to 2, I from 1 to 3, and J from 1 to 3.

(2) Format: Use type II. Each K needs two data cards with eight values on the first card and the last value on the first ten columns of the second card. No data card is needed if LOP = 1.

(3) Description:

ZC(K, I, J) is the jth coordinates of the ith point on the kth plane. K = 1 for top boundary plane and 2 for bottom boundary plane. J = 1 for x coordinate, 2 for y coordinate, and 3 for z coordinate. I is the point number.

2.3r z Coordinate of Each Layer Plane at Center of Well

(1) Variable: XZ(I); where I is from 1 to NZ.

(2) Format: Use type II.

(3) Description:

XZ(I) is the z coordinate for the ith layer plane at the center of well.

2.3s Number of Nodes on Each Radial Line

(1) Variable: ND(I); where I is from 1 to NR.

(2) Format: Use type I. No data card is needed if NTB = 0.

(3) Description:

ND(I) is the number of nodes on the ith radial line.

2.3t Radial Distances from Center of Well to Each Node

(1) Variable: R(I, J); where I is from 1 to NR, and J from 1 to ND(I).

(2) Format: Use type II. Starting from I = 1, punch ND(I) values of radial distances on data cards with eight values in each card. Each new value of I should start with a new card.

(3) Description:

R(I, J) is the radial distance from the center of well to the jth node on the ith radial line.

2.3u Angle Between Each Radial Line and First Radial Line

(1) Variable: O(I); where I is from 1 to NR.

(2) Format: Use type II.

(3) Description:

O(I) is the angle in radians between the ith radial line and the first radial line.

2.3v Permeabilities of Non-standard Elements

(1) Variable: PERM(I, J); where I is from 1 to NELEM, and J from 1 to 3.

(2) Format: Use type II. Three values of permeabilities are punched in one card. No data card is needed if NM = 0 or KP = 0.

(3) Description:

PERM(I, J) is the permeability as a ratio to the basic value for the ith element in the jth direction. I is the element number to be determined automatically by the computer. J = 1 for x direction, 2 for y direction, and 3 for z direction. The cards should be placed in the same order as those in section 2.3k, so that the permeabilities and the location will correspond.

2.3w Specific Storage Other than Basic Value

(1) Variable: STOR(I); where I is from 1 to NELEM.

(2) Format: Use type II. Only one value of specific storage is punched per card. No data card is needed if NM = 0 or KS = 0.

(3) Description:

STOR(I) is the specific storage of the ith element as a ratio to the basic value. The cards should be placed in the same order as those in section 2.3l, so that the specific storage and the location will correspond.

2.4 Summary of Required Cards

The deck setup for executing the program on IBM 360/65 is shown in Figure A12.

```
//jobname JOB account no., 'programmer's id', REGION=268K
//stepname EXEC FORTGCLP, TIME.GO=(min,sec)
//FORT.SYSIN DD *
                        Fortran source deck
/*
//GO.FT01F001 DD SPACE=(TRK,(20,10)),
// DCB=(RECFM=VSB,BLKSIZE=7294,LRECL=6000,BUFNO=1),UNIT=SYSDA
//GO.FT02F001 DD SPACE=(TRK,(20,10)),
// DCB=(RECFM=VSB,BLKSIZE=7294,LRECL=6000,BUFNO=1),UNIT=SYSDA
//GO.FT03F001 DD SPACE=(TRK,(20,10)),
// DCB=(RECFM=VSB,BLKSIZE=7294,LRECL=6000,BUFNO=1),UNIT=SYSDA
//GO.FT04F001 DD SPACE=(TRK,(20,10)),
// DCB=(RECFM=VSB,BLKSIZE=7294,LRECL=6000,BUFNO=1),UNIT=SYSDA
//GO.FT08F001 DD UNIT=2314,SPACE=(288,(500,40))
//GO.SYSIN DD *
                        Input data
/*
```

Figure A12. Deck setup for a FORTRAN source program.

If no plotting is desired, a convenient way to save time and eliminate the handling of plotting tape is to add a card "//GO.PLOTTAPE DD DUMMY" right after the first /* card. The jobname and stepname are 1-8 characters long. For problems involving large number of simultaneous equations, the block size of disks 1, 2, 3 and 4 should be made as large as possible, so a maximum size of 7294 is assigned. The block size of disk 8 for storing the element stiffness matrix is 288.

The total number of cards required for JCL (Job Control Language), main program, and subprograms, excluding all comment cards, is tabulated below

Type of Decks	Number of Cards
JCL	15
Main Program	300
Subprogram NODP	201
Subprogram PLANE	69
Subprogram BAPTI	111

Subprogram STIFI	84
Subprogram FEM	125
Subprogram TETRA	67
Subprogram FASEI	265
Subprogram SESOL	230
Subprogram PHPLOT	289
Total	1,756

2.5 Program Output

Two types of output can be obtained from the program, one is the general numerical output and the other is the plotting output. Both of these outputs have complete headings and should be self-explanatory. An itemized synopsis for each type of output is listed below.

2.5a Numerical Output

(1) A listing of major input data.

Major input data are printed out to insure that they are correct. They are printed out using the same format as the read statement except that some explanatory headings are added.

(2) A listing of literal output to indicate the type of aquifer and well considered.

(3) Computed output for checking purpose.

These output data are not part of the solution but can be used to check the correctness of some intermediate parameters which lead to the final solution. These parameters include nodal number, element number, total number of nodes, total number of equations, number of equations in each partition, and half band width. The reason that the total number of equations is not equal to the total number of nodes is because subprogram SESOL requires equal number of equations in each partition. If the total number of node is not divisible by the total number of partitions, fictitious equations are added to make each partition equal.

(4) Final solution on dimensionless or real drawdown at given dimensionless or real time.

(5) Execution time in miliseconds for solving simultaneous equations and the total time from the beginning.

2.5b Plotting Output

(1) A plane view of the aquifer and the finite element configuration.

(2) The drawdown at the centroid of each triangle or quadrilateral on a given layer plane at a given time.

The drawdown and time can be either dimensionless or real. Depending on the size of element, the drawdown will be printed either inside the element or outside the plan view of the aquifer as described in section 1.8i.

2.6 Off-Line Error Messages

Mistakes in input data or dimension statements will cause the computer to stop or yield erratic results. Be sure to check all input data cards carefully and make the size of all two dimensional arrays exactly the same as stipulated in section 2.7, except those in the common statement. The variables in the common block have been assigned a large dimension and, unless the dimension is exceeded, it is not necessary to change the array size. The size of all one dimensional arrays should be equal to or greater than that specified in section 2.7. If mistakes do occur, the following three error messages may be generated:

(1) If the dimensionless drawdown at the first nodal point, i.e. at the center of well on the top boundary, is greater than 100, the program will stop and print out the value of drawdown. When this occurs, the user should check the input data because it is unreasonable that the dimensionless drawdown be this large.

(2) If a diagonal element of [C] matrix is zero, the program will stop and print out the equation number in which zero diagonal occurs. [C] matrix is supposed to be positive and definite, and none of its diagonal elements should be zero.

(3) If a diagonal element of [C] matrix is negative, a warning message indicating the equation number and the value of the diagonal will be printed and the program will continue.

2.7 Variable Definitions

The following is a list of symbols used in the computer program together with their meaning and dimensions.

Notation	Description	Dimension Statement
AN(I)	Alphabetical characters used for plotting.	I = 20
BK	Basic value of permeability.	
BL	Basic value of length.	
BQ	Basic value of discharge.	
BS	Basic value of specific storage.	
C(I)	Overall stiffness matrix for each partition.	I = (MA+2) * NEQB
DATA(I)	Temporary array used in subroutine PLOTS.	I = 1024
EF(I, J)	Temporary array used in subprogram FASEI. I = NDP for infinitesimal wells; I = NDP + NZ * No. of nodes on loop 2 for finite wells; J = NZ * (1 + No. of nodes on loop 2 + NFW * No. of nodes on loop 3).	See description.
F(I, J)	Temporary array related to nodal discharge.	I = NEQB, J = 2
FQ(I)	Discharge at node i, positive for discharge and negative for recharge.	I = Max{KP, KS, JD, IE, KD, KH}
FT(I)	Temporary array used in subprogram FASEI.	I = NO
G(I)	Temporary array used in subprogram SESOL.	I = (MA+2) * NEQB
H(I, J)	Stiffness matrix of six- or eight-node element.	I = 36, J = 2
HH(K, I, J, L)	Stiffness matrix of tetrahedron	K = 10, I =4 J = 4, L = 2
IE	Total number of special elements.	
IH	Total number of nodes on a plan view.	
IS(I)	Type of special element.	I = Max{KP, KS, JD, IE, KD, KH}
JD	Total number of nodes at which drawdowns are to be printed; assgin 0 if drawdowns at all nodes are to be printed.	

Notation	Description	Dimension Statement
KA(I)	Temporary array indicating physically similar elements.	I = NELEM
KD	Total number of nodes at which discharges are given.	
KH	Total number of nodes at which drawdowns are always zero.	
KP	Total number of elements the permeabilities of which are different from those of the standard element.	
KS	Total number of elements the specific storage of which is different from the basic value.	
LOP	Orientation of top and bottom boundaries.	
MA	Half band width of simultaneous equations.	
MAXA(I)	Temporary array used in subprogram SESOL.	I = MA-1 + NEQB
MD	Total number of layer planes to be plotted.	
ME(I)	Layer plane number to be plotted.	I = MD
MG(I, J)	Temporary array used in subprogram PHPLOT.	I = NR, J = NP
MO(I, J)	Temporary array for nodal number on a plan view.	I = NP, J = NR
MPT	Total number of pumping times to be plotted.	
MT(I)	Time in miliseconds spent by the computer.	I = 3
NB	Total number of subdivisions.	
NCOLN	Type of well discharge assumed; assign 1 if discharge is uniform along the well bore, otherwise assign 2.	
ND(I)	Total number of nodes on each radial line.	I = NR
NDP	Total number of layer planes penetrated or tapped by the well.	
NELEM	Total number of elements.	
NEND(I)	Last element number in each partition.	I = NPART
NEQ	Total number of simultaneous equations.	
NEQB	Total number of equations per partition.	

Notation	Description	Dimension Statement
NFG	Type of aquifer; assign 0 for full aquifer and 1 for partial aquifer.	
NFW	Type of well; assign 0 for infinitesimal well and 1 for finite well.	
NH(I)	Nodal number at which drawdown is always zero.	I = Max{KP, KS, JD, IE, KD, KH}
NM	Type of aquifer; assign 0 for homogeneous aquifer and 1 for nonhomogeneous aquifer.	
NND(I)	Total number of elements per layer between two radial lines.	I = NR
NNP(I, J) NNR(I, J) NNZ(I, J)	Location of element or node; NNP for loop number, NNR for radial line number, and NNZ for layer plane number.	I = Max{KP, KS, JD, IE, KD, KH}
NO	Total number of nodal points.	
NOD(I, J)	Nodal number.	I = NELEM, J = 8
NP	Maximum number of nodes on a radial line.	
NPA	Number of partitions actually employed in the first time interval.	
NPART	Total number of partition.	
NQ(I)	Nodal number at which discharge is given.	I = Max{KP, KS, JD, IE, KD, KH}
NR	Total number of radial lines.	
NREAL	Type of data; assign 0 if time and drawdown are dimensionless, and 1 if they are real.	
NS(I)	Loop number from which the partial radial lines for the ith subdivision start.	I = NB
NSTART(I)	First element number in each partition.	I = NPART
NT	Total number of times at which drawdowns are computed including zero time.	
NTB	Type of boundary; assign 0 if total number of nodes in any radial line is equal to NP, otherwise assign 1.	
NWD(I)	Nodal number at which drawdown is to be printed.	I = JD
NZ	Total number of layer planes.	

Notation	Description	Dimension Statement
O(I)	Angle in radians between each radial line and the first radial line.	I = NR
PERM(I, J)	Permeabilities of element i in x, y and z directions.	I = NELEM, J = 3
R(I, J)	Radial distance from the center of well to each loop.	I = NR, J = NP
RR	Maximum radial distance from the center of well to the boundary of aquifer.	
S(I, J)	Drawdown at each node.	I = NEQ, J = 2
STOR(I)	Specific storage of element i.	I = NELEM
SK(I)	Permeability ratios in x, y and z directions of a standard element.	I = 3
T(I)	Time since pumping started.	I = NT
TP(I)	Pumping time at which drawdowns are plotted.	I = MPT
X(I, J)	Cartesian coordinates of each node in an element with respect to its centroid.	I = NO, J = 3
XM(I)	Temporary array used in subprogram PHPLOT.	I = IH
XZ(I)	z coordinate of each layer plane at the center of well.	I = NZ
YM(I)	Temporary array used in subprogram PHPLOT.	I = IH
Z(I, J)	Cartesian coordinates of each node.	I = NO, J = 3
ZC(K, I, J)	Cartesian coordinates of three points on each of the top and bottom boundaries.	K = 2, I = J = 3

2.8 Sample Problems

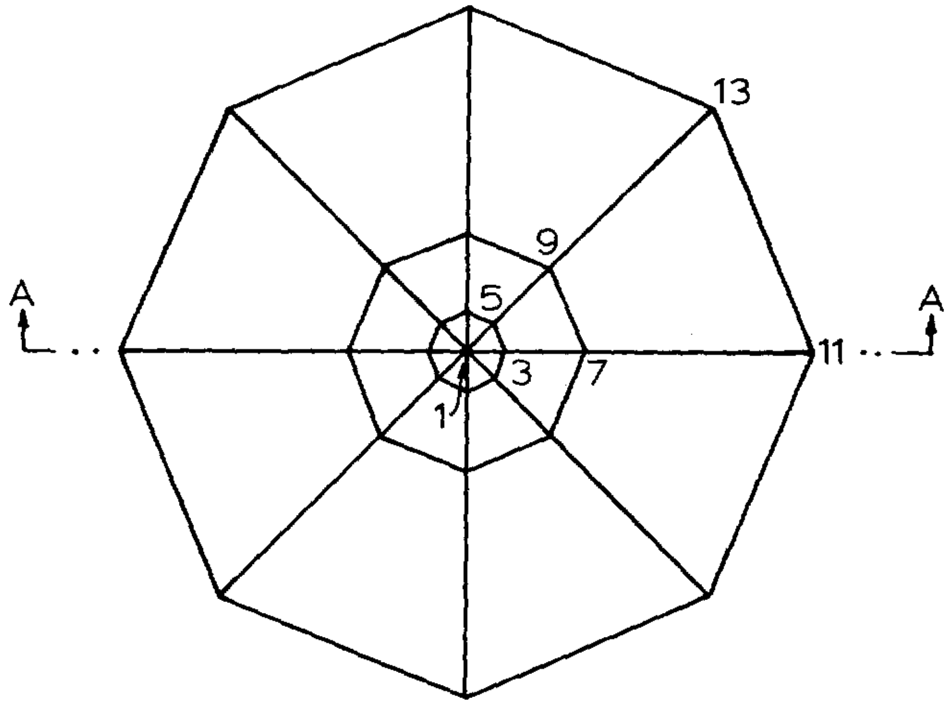
To illustrate the application of this program, especially the preparation of dimension statements and input data, two sample problems are presented. These two problems are quite different not only in complexities but also in basic assumptions. Following is a list of the differences between the two problems:

PROBLEM 1	PROBLEM 2
Infinitesimal well, NFW = 0	Finite well, NFW = 1
Fully penetrating well, NDP = NZ	Partially penetrating well, NDP < NZ
Uniform discharge along well, NCOLN = 1	Nonuniform discharge, NCOLN = 2
Homogeneous aquifer, NM = 0	Nonhomogeneous aquifer, NM = 1
Isotropic aquifer, SK(1) = SK(2) = SK(3) = 1	Anisotropic aquifer, SK(1) = SK(2) = 1, SK(3) $\neq$ 1
Partial aquifer, NFG = 1	Full aquifer, NFG = 0
Horizontal top and bottom boundaries, LOP = 1	Arbitrary top and bottom boundaries LOP = 3
Closed aquifer with no discharge, recharge or zero drawdown at any point, KD = 0, KH = 0.	Open aquifer with discharge, recharge, or zero drawdown at a specified number of nodes, KD $\neq$ 0, KH $\neq$ 0.
Same number of nodes on all radial lines, NTB = 0.	Number of nodes not the same on all radial lines, NTB = 1.
No subdivision, NB = 0.	With subdivision, NB $\neq$ 0.
No special elements, IE = 0.	With special elements, IE $\neq$ 0.
Number of equations = number of nodes.	Number of equations > number of nodes.
Dimensionless drawdown and time, NREAL = 0.	Real drawdown and time, NREAL = 1.
Printout of drawdown at all nodes, JD = 0.	Printout of drawdown only at given nodes, JD $\neq$ 0.
No plotting, MD = 0, MPT = 0.	With plotting, MD $\neq$ 0, MPT $\neq$ 0.

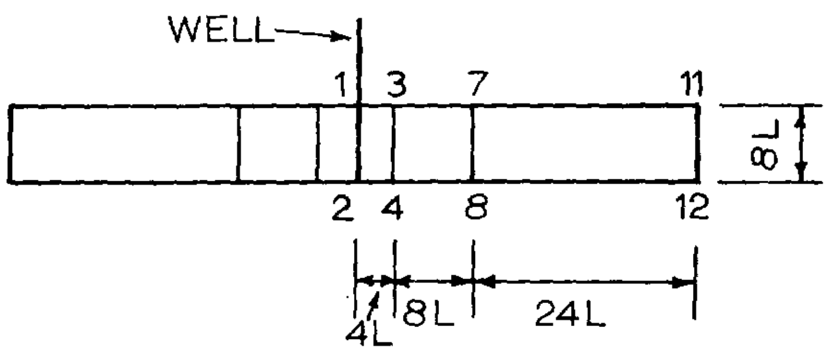
It should be noted that the purpose of these two problems is to provide complete input and output which permit the user to check a trial run on his computer against the sample data. They are certainly not realistic problems and only the first three dimensionless time increments are used.

2.8a Sample Problem 1

Figure A13 shows the plan view and cross section of an aquifer which is divided into finite elements. Because of axisymmetry, each radial plane can be considered as an impervious boundary, so only a slice of the aquifer is needed for analysis. For a well penetrating fully a horizontal aquifer of uniform thickness,



(a) PLAN VIEW



(b) SECTION A-A

Figure A13. A fully penetrating well in an axisymmetric aquifer.

the drawdown is independent of depth, so only one layer of finite elements is needed, and the discharge along the well can be taken as uniform. Assuming that the well is infinitesimal, the aquifer is homogeneous and isotropic, and the water is pumped out at a constant rate, the problem now on hand is to determine the dimensionless drawdown at a dimensionless time of 2.0.

The input data cards are listed as follows:

Card	Data									
1	NP	NR	NZ	NPART	NCOLN	NT	NFG	NM	NFW	JD	LOP
	4	2	2	1	1	4	1	0	0	0	1
	NDP	KP	KS	NTB	NB						
	2	0	0	0	0						
2	IE	NPA	MD	REAL	KD	KH	MPT				
	0	1	0	0	0	0	0				
3	BK	BS	BQ	BL	SK(1)	SK(2)	SK(3)				
	1.000E 00	1.000E 00	1.00000	1.00000	1.00000	1.00000	1.00000				
4	NND(1)										
	3										
5	T(1)	T(2)	T(3)	T(4)							
	0.00000	0.50000	1.00000	2.00000							
6	XZ(1)	XZ(2)									
	8.00000	0.00000									
7	R(1,1)	R(1,2)	R(1,3)	R(1,4)							
	0.00000	4.00000	12.00000	36.00000							
8	R(2,1)	R(2,2)	R(2,3)	R(2,4)							
	0.00000	4.00000	12.00000	36.00000							
9	O(1)	O(2)									
	0.00000	0.78540									

The complete output is shown in pages 86 to 88. Note that when the drawdown is small, it may turn out to be negative. This occurs only at the early stage of pumping and is due to the error inherent in the finite element method. The drawdown should be considered zero rather than negative.

The program requires a compiling time of 71 sec. The execution time, as printed in the output, is less than 4 sec.

SOLUTIONS OF THE UNSTEADY FLOW TOWARD WELLS

NP=	4	NR=	2	NZ=	2	NT=	4	NM=	0
JD=	0	KP=	0	KS=	0	NB=	0	IE=	0
MD=	0	KD=	0	KH=	0	NFW=	0	LOP=	1
NDP=	2	NTB=	0	NPA=	1	NFG=	1	MPT=	0
NPART=	1	NCOLN=	1	NREAL=	0				

INFINITESIMAL WELL

UNIFORM DISCHARGE ALONG THE WELL BORE

WELL PENETRATES AQUIFER TO POINT NUMBER: 2

THE WELL IS FULLY PENETRATING

AQUIFER IS HOMOGENEOUS THROUGHOUT

BASIC VALUES OF THE PARAMETERS :

PERMEABILITY	0.100E 01
SP. STORAGE	0.100E 01
DISCHARGE	1.00000
LENGTH	1.00000

PERMEABILITY RATIOS IN X, Y AND Z DIRECTIONS OF A STANDARD ELEMENT ARE :

1.00000 1.00000 1.00000

NO. OF ELEMENTS ON THE LEFT OF EACH RADIAL LINE IS:

3

LAYER HEIGHTS AT THE CENTER OF THE WELL ARE :

8.00000 0.0

RADIAL DISTANCES OF THE NODES FROM THE ORIGIN ON EACH RADIAL LINE:

RADIAL DISTANCES ON RADIAL LINE (1) :

0.0 4.00000 12.00000 36.00000

RADIAL DISTANCES ON RADIAL LINE (2) :

0.0 4.00000 12.00000 36.00000

ANGLES BETWEEN RADIAL LINES AND FIRST RADIAL LINE:

0.0 0.78540

TOTAL NUMBER OF NODES= 14
TOTAL NUMBER OF ELEMENTS= 3
TOTAL NUMBER OF EQUATIONS= 14
NUMBER OF EQUATIONS IN EACH PARTITION= 14
MAXIMUM HALF BAND WIDTH= 8

STARTING AND ENDING ELEMENTS IN EACH PARTITION ARE:

PARTITION NO.= 1 NSTART= 1 NEND= 3

DIMENSIONLESS VALUES

TIME= 0.50000

NOD	DRAWDOWN	NOD	DRAWDOWN	NOD	DRAWDOWN
1	0.62774837E-01	2	0.62774956E-01	3	-0.37953537E-02
4	-0.37953611E-02	5	-0.37953556E-02	6	-0.37953649E-02
7	0.29573590E-03	8	0.29573566E-03	9	0.29573636E-03
10	0.29573590E-03	11	-0.11778985E-03	12	-0.11778995E-03
13	-0.11778990E-03	14	-0.11779007E-03		

TIME PERIOD= 1 TIME FOR EQ. SOLVER= 768 TOTAL TIME SPENT= 1675

DIMENSIONLESS VALUES

TIME= 1.00000

NOD	DRAWDOWN	NOD	DRAWDOWN	NOD	DRAWDOWN
1	0.11075258E 00	2	0.11075264E 00	3	-0.55699274E-02
4	-0.55699199E-02	5	-0.55699088E-02	6	-0.55699237E-02
7	0.41346485E-03	8	0.41346555E-03	9	0.41346601E-03
10	0.41346508E-03	11	-0.16367261E-03	12	-0.16367272E-03
13	-0.16367261E-03	14	-0.16367319E-03		

TIME PERIOD= 2 TIME FOR EQ. SOLVER= 927 TOTAL TIME SPENT= 2635

DIMENSIONLESS VALUES

TIME= 2.00000

NOD	DRAWDOWN	NOD	DRAWDOWN	NOD	DRAWDOWN
1	0.17735022E 00	2	0.17734998E 00	3	-0.52489527E-02
4	-0.52489564E-02	5	-0.52489303E-02	6	-0.52489489E-02
7	0.32995595E-03	8	0.32995688E-03	9	0.32995641E-03
10	0.32995618E-03	11	-0.12774288E-03	12	-0.12774323E-03
13	-0.12774323E-03	14	-0.12774393E-03		

TIME PERIOD= 3 TIME FOR EQ. SOLVER= 922 TOTAL TIME SPENT= 3590

2.8b Sample Problem 2

The aquifer shown in Figure A11 is used as an example. The aquifer is nonhomogeneous with permeabilities and specific storage in the shaded region different from those in the unshaded region. Because there are more elements in the unshaded region, its specific storage and one of its permeabilities are used as basic values. At the same time as pumping starts, recharges occur at nodes 55, 56 and 57, discharges at nodes 99, 102 and 105, and zero drawdown at nodes 79 through 87.

The input data cards are listed below. To save space, only the first card in a group is listed, and the remainder can be found from the output data.

Card	Data										
1	NP	NR	NZ	NPART	NCOLN	NT	NFG	NM	NFW	JD	LOP
	5	12	3	2	2	4	0	1	1	8	3
	NDP	KP	KS	NTB	NB						
	2	18	18	1	1						
2	IE	NPA	MD	REAL	KD	KH	MPT				
	6	2	1	1	6	9	1				
3	BK	BS	BQ	BL	SK(1)	SK(2)	SK(3)				
	1.000E-04	1.000E-05	0.80000	1.00000	1.00000	1.00000	0.05000				
4	NNP(1,1)	NNR(1,1)	NNZ(1,1)								
	1	1	1								
	. . . .										
12	NS(1)										
	4										
13	NND(1)	NND(2)	NND(3)	NND(4)	NND(5)	NND(6)	NND(7)				
	4	4	4	4	4	2	3				
	NND(8)	NND(9)	NND(10)	NND(11)	NND(12)						
	4	2	4	4	4						
14	T(1)	T(2)	T(3)	T(4)							
	0.00000	0.50000	1.00000	2.00000							
15	RR										
	80.00000										
16	TP(1)										
	2.00000										
17	ME(1)										
	1										

18	NNP(1,2)	NNR(1,2)	NNZ(1,2)
	2	1	2

. . . .

36	NNP(1,3)	NNR(1,3)	NNZ(1,3)
	2	1	2

54	NNP(1,4)	NNR(1,4)	NNZ(1,4)	IS(1)
	4	6	1	3

. . . .

60	NNP(1,5)	NNR(1,5)	NNZ(1,5)	FQ(1)
	4	6	1	-0.06000

. . . .

66	NNP(1,6)	NNR(1,6)	NNZ(1,6)
	5	3	1

. . . .

75	ZC(1,1,1)	ZC(1,1,2)	ZC(1,1,3)	ZC(1,2,1)	ZC(2,2,2)
	0.00000	0.00000	40.00000	100.00000	0.00000
	ZC(2,2,3)	ZC(1,3,1)	ZC(1,3,2)		
	44.00000	0.00000	100.00000		

76	ZC(1,3,3)
	40.00000

. . . .

79	XZ(1)	XZ(2)	XZ(3)
	40.00000	30.00000	10.00000

80	ND(1)	ND(2)	ND(3)	ND(4)	ND(5)	ND(6)	ND(7)	ND(8)	ND(9)
	5	4	5	5	5	5	4	5	5
	ND(10)	ND(11)	ND(12)						
	5	5	5						

81	R(1,1)	R(1,2)	R(1,3)	R(1,4)	R(1,5)
	0.00000	10.00000	22.00000	40.00000	62.00000

. . . .

93	O(1)	O(2)	O(3)	O(4)	O(5)	O(6)	O(7)	O(8)
	0.00000	0.43633	0.87267	1.57080	2.18167	2.87980	3.52557	4.08408
	O(9)	O(10)	O(11)	O(12)				
	4.46805	4.97420	5.41053	5.84687				

94	PERM(LK,1)	PERM(LK,2)	PERM(LK,3)
	2.50000	1.20000	0.05000

. . . .

112	STOR(LK) 1.20000
	. . . .
129	1.2000 (last card)

The complete output is shown in pages 93 to 98. As shown in the output, the total execution time for this sample problem is about 29 sec, excluding 71 sec of compiling time. The plotted output is shown in Figure A14. It should be reiterated that the purpose of the sample problems is to show the type of input and output data only. Because of the unrealistic assumptions regarding the well and the boundaries, the computer output may not seem reasonable. However, the computer program was tested for many realistic cases, and reasonable results were obtained.

AVERAGE REAL DRAWDOWN FOR REAL TIME= 0.2

A	1.41
B	1.41
C	1.41
D	1.41

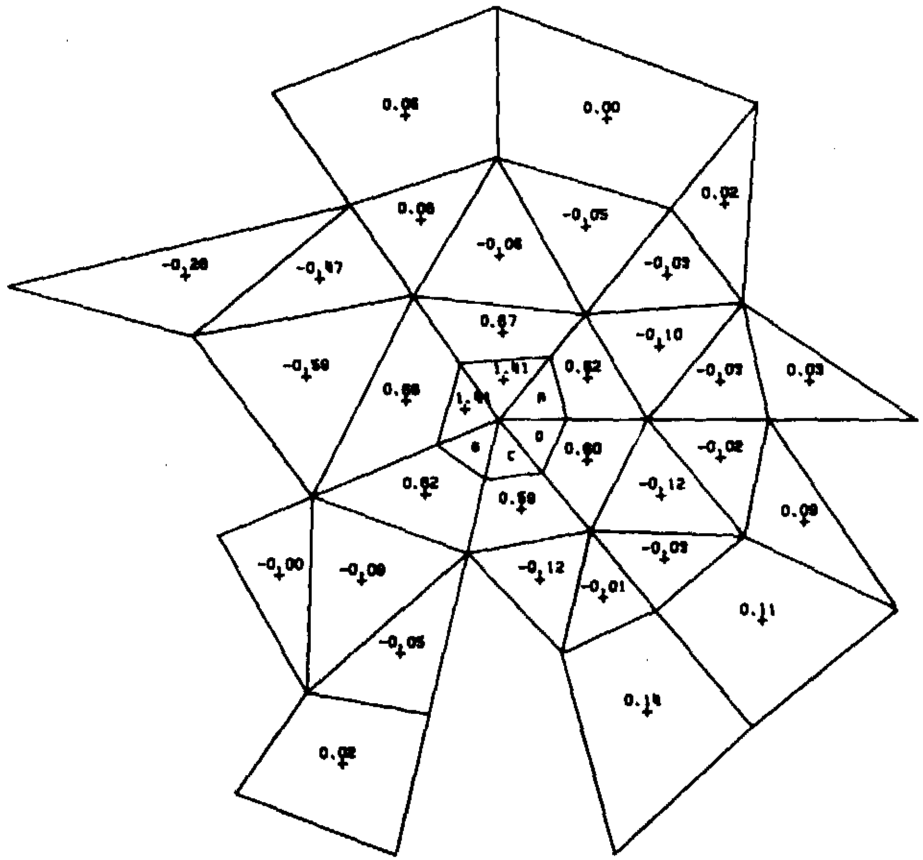


Figure.A14. Plotting of sample problem 2.

SOLUTIONS OF THE UNSTEADY FLOW TOWARD WELLS

NP=	5	NR=	12	NZ=	3	NT=	4	NM=	1
JD=	8	KP=	18	KS=	18	NB=	1	IE=	6
MD=	1	KD=	6	KH=	9	NFW=	1	LOP=	3
NDP=	2	NTB=	1	NPA=	2	NFG=	0	MPT=	1
NPART=	2	NCOLN=	2	NREAL=	1				

FINITE WELL

NONUNIFORM DISCHARGE ALONG THE WELL BORE

WELL PENETRATES AQUIFER TO POINT NUMBER: 2

THE WELL IS PARTIALLY PENETRATING

AQUIFER IS NONHOMOGENEOUS

BASIC VALUES OF THE PARAMETERS :

PERMEABILITY	0.100E-03
SP. STORAGE	0.100E-04
DISCHARGE	0.80000
LENGTH	1.00000

PERMEABILITY RATIOS IN X, Y AND Z DIRECTIONS OF A STANDARD ELEMENT ARE :

1.00000 1.00000 0.05000

NO. OF SUBDIVISION= 1 ARC NO.= 4

NO. OF ELEMENTS ON THE LEFT OF EACH RADIAL LINE IS:

4 4 4 4 4 2 3 4 2 4 4 4

MAX. RADIAL DISTANCE FROM THE CENTER : 80.00000

TIMES TO BE PLOTTED ARE :
2.00000

PLANES TO BE PLOTTED ARE:
1

LOCATIONS AND TYPES OF SPECIAL ELEMENTS

NNP	NNR	NNZ	IS
4	6	1	3
4	6	2	3
5	5	1	2
5	5	2	2
5	12	1	1
5	12	2	1

LOCATIONS AND DISCHARGES OR RECHARGES OF THE NODES:

LOCATION			DISCHARGE	LOCATION			DISCHARGE
NNP	NNR	NNZ	FQ	NNP	NNR	NNZ	FQ
4	6	1	-0.06000	4	6	2	-0.08000
4	6	3	-0.06000	5	10	3	0.10000
5	11	3	0.10000	5	12	3	0.10000

LOCATIONS OF THE NODES WITH ZERO DRAWDOWNS:

LOCATION			LOCATION			LOCATION		
NNP	NNR	NNZ	NNP	NNR	NNZ	NNP	NNR	NNZ
5	3	1	5	3	2	5	3	3
5	4	1	5	4	2	5	4	3
5	5	1	5	5	2	5	5	3

COORDINATES OF GIVEN THREE POINTS ON PLANE (1):

X	Y	Z
0.0	0.0	40.00000
100.00000	0.0	44.00000

```
        0.0                100.00000                40.00000

   COORDINATES OF GIVEN THREE POINTS ON PLANE (2):
          X                    Y                    Z

        0.0                  0.0                10.00000
      100.00000              0.0                 0.0
        0.0                100.00000            10.00000

   LAYER HEIGHTS AT THE CENTER OF THE WELL ARE :
     40.00000  30.00000  10.00000

   NUMBER OF NODAL POINTS ON EACH RADIAL LINE IS:
  5    4    5    5    5    5    4    5    5    5    5    5

   RADIAL DISTANCES OF THE NODES FROM THE ORIGIN ON EACH RADIAL LINE:

   RADIAL DISTANCES ON RADIAL LINE ( 1) :
          0.0       10.00000  22.00000  40.00000  62.00000
   RADIAL DISTANCES ON RADIAL LINE ( 2) :
          0.0        0.0       0.0      40.00000
   RADIAL DISTANCES ON RADIAL LINE ( 3) :
          0.0       12.00000  20.00000  40.00000  60.00000
   RADIAL DISTANCES ON RADIAL LINE ( 4) :
          0.0        0.0       0.0      38.00000  60.00000
   RADIAL DISTANCES ON RADIAL LINE ( 5) :
          0.0       10.00000  22.00000  38.00000  58.00000
   RADIAL DISTANCES ON RADIAL LINE ( 6) :
          0.0        0.0       0.0      47.00000  75.00000
   RADIAL DISTANCES ON RADIAL LINE ( 7) :
          0.0       10.00000  30.00000  45.00000
   RADIAL DISTANCES ON RADIAL LINE ( 8) :
          0.0        0.0       0.0      49.00000  67.00000
   RADIAL DISTANCES ON RADIAL LINE ( 9) :
          0.0        9.00000  20.00000  44.00000  65.00000
   RADIAL DISTANCES ON RADIAL LINE (10) :
          0.0        0.0       0.0      35.00000  65.00000
```

RADIAL DISTANCES ON RADIAL LINE (11) :

0.0 10.00000 21.00000 36.00000 58.00000

RADIAL DISTANCES ON RADIAL LINE (12) :

0.0 0.0 0.0 40.00000 65.00000

ANGLES BETWEEN RADIAL LINES AND FIRST RADIAL LINE:

0.0 0.43633 0.87267 1.57080 2.18167 2.87980 3.52557 4.08408
4.46805 4.97420 5.41053 5.84687

PERMEABILITIES AND SPECIFIC STORAGES OF THE NON-STANDARD ELEMENTS :

NNP	NNR	NNZ	ELEM. NO.	PERMEABILITY OR SPECIFIC STORAGE			
2	1	2	2	PERM	XK= 2.50000	YK= 1.20000	ZK= 0.05000
2	3	2	4	PERM	XK= 2.50000	YK= 1.20000	ZK= 0.05000
2	5	2	6	PERM	XK= 2.50000	YK= 1.20000	ZK= 0.05000
2	7	2	8	PERM	XK= 2.50000	YK= 1.20000	ZK= 0.05000
2	9	2	10	PERM	XK= 2.50000	YK= 1.20000	ZK= 0.05000
2	11	2	12	PERM	XK= 2.50000	YK= 1.20000	ZK= 0.05000
3	1	1	13	PERM	XK= 2.50000	YK= 1.20000	ZK= 0.05000
3	1	2	14	PERM	XK= 2.50000	YK= 1.20000	ZK= 0.05000
3	3	1	15	PERM	XK= 2.50000	YK= 1.20000	ZK= 0.05000
3	3	2	16	PERM	XK= 2.50000	YK= 1.20000	ZK= 0.05000
3	5	1	17	PERM	XK= 2.50000	YK= 1.20000	ZK= 0.05000
3	5	2	18	PERM	XK= 2.50000	YK= 1.20000	ZK= 0.05000
3	7	1	19	PERM	XK= 2.50000	YK= 1.20000	ZK= 0.05000
3	7	2	20	PERM	XK= 2.50000	YK= 1.20000	ZK= 0.05000
3	9	1	21	PERM	XK= 2.50000	YK= 1.20000	ZK= 0.05000
3	9	2	22	PERM	XK= 2.50000	YK= 1.20000	ZK= 0.05000
3	11	1	23	PERM	XK= 2.50000	YK= 1.20000	ZK= 0.05000
3	11	2	24	PERM	XK= 2.50000	YK= 1.20000	ZK= 0.05000
2	1	2	2	STOR		1.20000	
2	3	2	4	STOR		1.20000	
2	5	2	6	STOR		1.20000	
2	7	2	8	STOR		1.20000	
2	9	2	10	STOR		1.20000	
2	11	2	12	STOR		1.20000	

3	1	1	13	STOR	1.20000
3	1	2	14	STOR	1.20000
3	3	1	15	STOR	1.20000
3	3	2	16	STOR	1.20000
3	5	1	17	STOR	1.20000
3	5	2	18	STOR	1.20000
3	7	1	19	STOR	1.20000
3	7	2	20	STOR	1.20000
3	9	1	21	STOR	1.20000
3	9	2	22	STOR	1.20000
3	11	1	23	STOR	1.20000
3	11	2	24	STOR	1.20000

TOTAL NUMBER OF NODES= 105
TOTAL NUMBER OF ELEMENTS= 74
TOTAL NUMBER OF EQUATIONS= 106
NUMBER OF EQUATIONS IN EACH PARTITION= 53
MAXIMUM HALF BAND WIDTH= 65

STARTING AND ENDING ELEMENTS IN EACH PARTITION ARE:

PARTITION NO.= 1 NSTART= 1 NEND= 74
PARTITION NO.= 2 NSTART= 36 NEND= 74

REAL VALUES

DIMENSIONLESS TIME= 0.50000 OR REAL TIME= 0.05000

NOD	NNP	NNR	NNZ	DRAWDOWN	NOD	NNP	NNR	NNZ	DRAWDOWN
1	1	1	1	0.48780435E 00	4	2	1	1	0.48780435E 00
13	2	7	1	0.48780435E 00	22	3	1	1	-0.74386597E-01
31	3	7	1	-0.51498413E-01	40	4	1	1	0.19550316E-01
58	4	7	1	0.95963404E-02	76	5	1	1	-0.73015690E-02

TIME PERIOD= 1 TIME FOR EQ. SOLVER= 6327 TOTAL TIME SPENT= 13376

REAL VALUES

DIMENSIONLESS TIME= 1.00000 OR REAL TIME= 0.10000

NOD	NNP	NNR	NNZ	DRAWDOWN	NOD	NNP	NNR	NNZ	DRAWDOWN
1	1	1	1	0.81014621E 00	4	2	1	1	0.81014621E 00
13	2	7	1	0.81014621E 00	22	3	1	1	-0.11825556E 00
31	3	7	1	-0.69618165E-01	40	4	1	1	0.35524365E-01
58	4	7	1	0.94771385E-02	76	5	1	1	-0.88512860E-02

TIME PERIOD= 2 TIME FOR EQ. SOLVER= 7104 TOTAL TIME SPENT= 20515

REAL VALUES

DIMENSIONLESS TIME= 2.00000 OR REAL TIME= 0.20000

NOD	NNP	NNR	NNZ	DRAWDOWN	NOD	NNP	NNR	NNZ	DRAWDOWN
1	1	1	1	0.14090528E 01	4	2	1	1	0.14090528E 01
13	2	7	1	0.14090528E 01	22	3	1	1	-0.18978113E 00
31	3	7	1	-0.93460083E-01	40	4	1	1	0.63776910E-01
58	4	7	1	0.75697899E-02	76	5	1	1	-0.92089176E-02

TIME PERIOD= 3 TIME FOR EQ. SOLVER= 7093 TOTAL TIME SPENT= 29481

```
C                                                                            C
C****************************************************************************C
C                                                                            C
C                      PART III  LIST OF "UNFLAW" PROGRAM                    C
C                                                                            C
C          THIS THREE DIMENSIONAL FINITE ELEMENT PROGRAM IS USED FOR CALCULATING C
C       THE UNSTEADY DRAWDOWNS AROUND AN ARTESIAN WELL.  THE AQUIFER MAY BE   C
C       NONHOMOGENEOUS AND ANISOTROPIC WITH DISCHARGES, RECHARGES, OR ZERO    C
C       DRAWDOWNS AT VARIOUS POINTS SPECIFIED.  THE WELL MAY BE FINITE OR     C
C       INFINITESIMAL IN RADIUS, FULLY OR PARTIALLY PENETRATING, WITH UNIFORM C
C       OR NONUNIFORM DISCHARGE ALONG THE WELL BORE.  FOR A FINITE WELL ONLY  C
C       THE CASE OF NONUNIFORM DISCHARGE IS CONSIDERED.  THE TOP AND BOTTOM   C
C       BOUNDARY PLANES OF THE AQUIFER ARE ARBITRARY.  THE FINAL RESULTS CAN  C
C       BE EITHER DIMENSIONAL (REAL) OR DIMENSIONLESS VALUES.  THE DIMENSION  C
C       STATEMENTS WERE USED FOR RUNNING SAMPLE PROBLEM NO. 2.  HOWEVER, 100  C
C       ELEMENTS AND 200 NODAL POINTS ARE ASSUMED, WHICH ARE GREATER THAN THE C
C       ACTUAL VALUES.  THIS PROGRAM WITH THE GIVEN DIMENSION STATEMENTS      C
C       REQUIRES A TOTAL STORAGE OF 214K.                                     C
C                                                                            C
C****************************************************************************C
C                                                                            C
      DIMENSION AN(20),MT(3),SK(3),DATA(1024),H(36,2),ZC(2,3,3)
      DIMENSION O(20),T(20),ND(20),XZ(20),NS(20),KA(100),ME(20)
      DIMENSION XM(100),YM(100),NWD(50),NND(20),NSTART(10),NEND(10)
      DIMENSION FT(200),C(5000),G(5000),MAXA(200),F(53,2),EF(20,39)
      DIMENSION IS(18),NH(18),NQ(18),NNP(18,6),NNR(18,6),NNZ(18,6)
      DIMENSION FQ(18),R(12,5),MG(12,5),MO(5,12),TP(20),STOR(100)
      DIMENSION PERM(100,3),NOD(100,8),S(200,2),X(200,3),Z(200,3)
      COMMON PERM,NOD,S,X,Z
      DEFINE FILE 8(74,288,L,III)
      DATA AN/'A','B','C','D','E','F','G','H','O','P','Q','R','S','T',
     *       'U','V','W','X','Y','Z'/
      CALL TIMER (0)
      WRITE (6,10)
   10 FORMAT ('1',15X,'SOLUTIONS OF THE UNSTEADY FLOW TOWARD WELLS',//)
```

```
C
C**********************************************************************C
C     INITIALIZE PLOT                                                  C
C**********************************************************************C
C
      CALL PLOTS(DATA,4096)
      CALL PLOT (0.0,-11.0,-3)
      CALL PLOT (0.0,0.2,-3)
C
C**********************************************************************C
C     READ IN GENERAL VALUES                                           C
C**********************************************************************C
C
      READ (5,20) NP,NR,NZ,NPART,NCOLN,NT,NFG,NM,NFW,JD,LOP,
     *     NDP,KP,KS,NTB,NB,IE,NPA,MD,NREAL,KD,KH,MPT
   20 FORMAT (16I5)
      WRITE (6,30) NP,NR,NZ,NT,NM,JD,KP,KS,NB,IE,MD,KD,KH,NFW,LOP,NDP,
     *    NTB,NPA,NFG,MPT,NPART,NCOLN,NREAL
   30 FORMAT (/,5X,'NP=',I5,5X,'NR=',I5,5X,'NZ=',I5,5X,'NT=',I5,5X,
     *     'NM=',I5,/,5X,'JD=',I5,5X,'KP=',I5,5X,'KS=',I5,5X,'NB=',I5,
     *     5X,'IE=',I5,/,5X,'MD=',I5,5X,'KD=',I5,5X,'KH=',I5,4X,'NFW=',
     *     I5,4X,'LOP=',I5,/,4X,'NDP=',I5,4X,'NTB=',I5,4X,'NPA=',I5,4X,
     *     'NFG=',I5,4X,'MPT=',I5,/,2X,'NPART=',I5,2X,'NCOLN=',I5,2X,
     *     'NREAL=',I5,/)
C
C**********************************************************************C
C     DEFINE THE TYPE AND THE CONDITIONS OF THE WELL AND THE AQUIFER   C
C          CONSIDERED                                                  C
C**********************************************************************C
C
      IF (NFW .GT. 0) GO TO 50
      WRITE (6,40)
   40 FORMAT (/,12X,'INFINITESIMAL WELL')
      GO TO 70
   50 WRITE (6,60)
```

```
   60 FORMAT (/,12X,'FINITE WELL')
   70 IF (NCOLN .GT. 1) GO TO 90
      WRITE (6,80)
   80 FORMAT (/,12X,'UNIFORM DISCHARGE ALONG THE WELL BORE')
      GO TO 110
   90 WRITE (6,100)
  100 FORMAT(/,12X,'NONUNIFORM DISCHARGE ALONG THE WELL BORE')
  110 WRITE (6,120) NDP
  120 FORMAT(/,12X,'WELL PENETRATES AQUIFER TO POINT NUMBER:',2X,I5)
      IF (NDP .LT. NZ) GO TO 140
      WRITE(6,130)
  130 FORMAT (/,12X,'THE WELL IS FULLY PENETRATING')
      GO TO 160
  140 WRITE(6,150)
  150 FORMAT(/,12X,'THE WELL IS PARTIALLY PENETRATING')
  160 IF (NM .GT. 0) GO TO 180
      WRITE(6,170)
  170 FORMAT(/,12X,'AQUIFER IS HOMOGENEOUS THROUGHOUT',/)
      GO TO 200
  180 WRITE (6,190)
  190 FORMAT (/,12X,'AQUIFER IS NONHOMOGENEOUS',/)
C
C*******************************************************************************C
C     READ IN BASIC VALUES FOR PERMEABILITY(BK),SPECIFIC STORAGE(BS),RATE      C
C          OF DISCHARGE(BQ),AND LENGTH(BL),AND ALSO PERMEABILITY RATIO IN      C
C          X,Y AND Z DIRECTIONS OF A STANDARD ELEMENT                          C
C*******************************************************************************C
C
  200 READ (5,210) BK,BS,BQ,BL,(SK(I),I=1,3)
  210 FORMAT (2E10.3,5F10.5)
      WRITE(6,220) BK,BS,BQ,BL,(SK(I),I=1,3)
  220 FORMAT (5X,'BASIC VALUES OF THE PARAMETERS :',/,8X,'PERMEABILITY',
     *  4X,E10.3,/,8X,'SP. STORAGE',5X,E10.3,/,8X,'DISCHARGE',7X,F10.5,
     * /,8X,'LENGTH',10X,F10.5,//,5X,'PERMEABILITY RATIOS IN X, Y AND Z
     *DIRECTIONS OF A STANDARD ELEMENT ARE :',/,3(5X,F10.5))
```

```
C
C*************************************************************************C
C      READ IN THE LOCATION OF EACH NODAL POINT AT WHICH DRAWDOWN IS TO BE  C
C           PRINTED OUT                                                    C
C*************************************************************************C
C
       IF (JD .EQ. 0) GO TO 240
       DO 230 I=1,JD
   230 READ (5,20) NNP(I,1),NNR(I,1),NNZ(I,1)
C
C*************************************************************************C
C      READ IN THE LOOP NUMBER AT WHICH THE SUBDIVISION OF THE ELEMENTS STARTS C
C*************************************************************************C
C
   240 IF (NB .EQ. 0) GO TO 270
       READ (5,20) (NS(I),I=1,NB)
       DO 250 I=1,NB
   250 WRITE(6,260) I,NS(I)
   260 FORMAT(/,5X,'NO. OF SUBDIVISION=',I5,5X,'ARC NO.=',I5)
C
C*************************************************************************C
C      READ IN TOTAL NUMBER OF ELEMENTS ON THE TOP PLANE OF EACH SLICE,OR   C
C           ON THE LEFT SIDE OF EACH RADIAL LINE                           C
C*************************************************************************C
C
   270 K=NR
       IF (NFG .GT. 0) K=NR-1
       READ (5,20) (NND(I),I=1,K)
       WRITE(6,280) (NND(I),I=1,K)
   280 FORMAT(/,5X,'NO. OF ELEMENTS ON THE  LEFT OF EACH RADIAL LINE IS:'
      *,/,5X,16I5)
C
C*************************************************************************C
C      READ IN TIME PERIODS MEASURED SINCE PUMPING STARTED                 C
C*************************************************************************C
```

```
C
      READ (5,290) (T(I),I=1,NT)
  290 FORMAT (6F13.5)
      IF (MD .EQ. 0 .OR. MPT .EQ. 0) GO TO 340
C
C**********************************************************************C
C     READ IN MAX. RADIAL DISTANCE MEASURED FROM THE CENTER OF THE WELL   C
C**********************************************************************C
C
      READ (5,300) RR
  300 FORMAT (F10.5)
      WRITE (6,310) RR
  310 FORMAT(/,5X,'MAX. RADIAL DISTANCE FROM THE CENTER :',2X,F10.5,/)
C
C**********************************************************************C
C     READ IN VALUE OF EACH ASSIGNED PUMPING TIME AT WHICH A PLOT OF THE  C
C         AVERAGE DRAWDOWN IS REQUIRED                                    C
C**********************************************************************C
C
      READ (5,290) (TP(I),I=1,MPT)
      WRITE(6,320) (TP(I),I=1,MPT)
  320 FORMAT (5X,'TIMES TO BE PLOTTED ARE :',/,5X,6(F13.5,2X))
C
C**********************************************************************C
C     READ IN NUMBER OF EACH ASSIGNED PLANE IN WHICH AVERAGE DRAWDOWN     C
C          IS TO BE PLOTTED                                               C
C**********************************************************************C
C
      READ (5,20) (ME(I),I=1,MD)
      WRITE(6,330)(ME(I),I=1,MD)
  330 FORMAT (/,5X,'PLANES TO BE PLOTTED ARE:',/,10X,12I5)
  340 IF (NM .EQ. 0) GO TO 380
C
C**********************************************************************C
C     READ IN LOCATIONS OF ELEMENTS HAVING PERMEABILITIES DIFFERENT FROM  C
```

```
C          THOSE OF THE STANDARD ELEMENT OR SPECIFIC STORAGES DIFFERENT FROM ONE C
C***********************************************************************************C
C
       IF (KP .EQ. 0) GO TO 360
       DO 350 I=1,KP
   350 READ (5,20) NNP(I,2),NNR(I,2),NNZ(I,2)
   360 IF (KS .EQ. 0) GO TO 380
       DO 370 I=1,KS
   370 READ (5,20) NNP(I,3),NNR(I,3),NNZ(I,3)
C
C***********************************************************************************C
C      READ IN LOCATIONS OF SPECIAL ELEMENTS                                        C
C***********************************************************************************C
C
   380 IF (IE .EQ. 0) GO TO 420
       WRITE(6,390)
   390 FORMAT(/,5X,'LOCATIONS AND TYPES OF SPECIAL ELEMENTS',/,8X,'NNP',
      *   5X,'NNR',5X,'NNZ',6X,'IS',/)
       DO 400 I=1,IE
       READ (5,20) NNP(I,4),NNR(I,4),NNZ(I,4),IS(I)
   400 WRITE(6,410) NNP(I,4),NNR(I,4),NNZ(I,4),IS(I)
   410 FORMAT(8X,I3,3(5X,I3))
C
C***********************************************************************************C
C      READ IN LOCATIONS OF POINTS AT WHICH DISCHARGES OR ZERO DRAWDOWNS ARE      C
C           GIVEN                                                                   C
C***********************************************************************************C
C
   420 IF (KD .EQ. 0) GO TO 470
       WRITE (6,430)
   430 FORMAT(/,5X,'LOCATIONS AND DISCHARGES OR RECHARGES OF THE NODES:',
      *   /,2(8X,'LOCATION',12X,'DISCHARGE'),/,2(5X,'NNP',3X,'NNR',3X,
      *   'NNZ',12X,'FQ',4X),/)
       DO 440 I=1,KD
   440 READ (5,450) NNP(I,5),NNR(I,5),NNZ(I,5),FQ(I)
```

```
  450 FORMAT (3I5,F10.5)
      WRITE(6,460) (NNP(I,5),NNR(I,5),NNZ(I,5),FQ(I),I=1,KD)
  460 FORMAT (2(5X,I3,2(3X,I3),8X,F10.5))
  470 IF (KH .EQ. 0) GO TO 510
      WRITE (6,480)
  480 FORMAT(/,5X,'LOCATIONS OF THE NODES WITH ZERO DRAWDOWNS:',/,8X,
     *   3('LOCATION',17X),/,3(5X,'NNP',3X,'NNR',3X,'NNZ',5X),/)
      DO 490 I=1,KH
  490 READ (5,20) NNP(I,6),NNR(I,6),NNZ(I,6)
      WRITE (6,500) (NNP(I,6),NNR(I,6),NNZ(I,6),I=1,KH)
  500 FORMAT (3(5X,I3,2(3X,I3),5X))
C
C*****************************************************************************C
C     READ IN INFORMATIONS FOR DETERMINING PLANE EQUATIONS AND COORDINATES OF  C
C       NODAL POINTS; IF LOP=1, READ IN Z COORDINATES OF ALL THE LAYER PLANES  C
C        ALONG THE CENTER OF WELL; IF LOP=2 OR 3, READ IN THE COORDINATES      C
C        OF ANY THREE POINTS ON THE TOP AND BOTTOM BOUNDARY PLANES, AND THE    C
C        Z COORDINATE OF ALL LAYER PLANES ALONG THE CENTER OF THE WELL         C
C     K=1 FOR TOP PLANE, AND K=2 FOR BOTTOM PLANE                              C
C*****************************************************************************C
C
  510 IF (LOP .EQ. 1) GO TO 560
      DO 550 K=1,2
      WRITE(6,520) K
  520 FORMAT(/,5X,'COORDINATES OF GIVEN THREE POINTS ON PLANE (',I1,'):'
     *,/,14X,'X',18X,'Y',18X,'Z',/)
      READ(5,530) ((ZC(K,I,J),J=1,3),I=1,3)
  530 FORMAT (8F10.5)
      WRITE(6,540) ((ZC(K,I,J),J=1,3),I=1,3)
  540 FORMAT(3(9X,F10.5))
  550 CONTINUE
  560 XNZ=NZ-1
      READ (5,570) (XZ(I),I=1,NZ)
  570 FORMAT (8F10.5)
      WRITE(6,580) (XZ(I),I=1,NZ)
```

```
  580 FORMAT(/,5X,'LAYER HEIGHTS AT THE CENTER OF THE WELL ARE :',/,5X,
     *       8F10.5)
C
C***********************************************************************C
C      READ IN NUMBER OF POINTS AND THEIR RADIAL DISTANCES IN EACH RADIAL C
C         LINE, AND ALSO READ IN EACH ANGLE FORMED BY EACH RADIAL LINE WITH C
C         RESPECT TO THE FIRST RADIAL LINE                               C
C***********************************************************************C
C
      IF (NTB .EQ. 0) GO TO 600
      READ (5,20) (ND(I),I=1,NR)
      WRITE(6,590) (ND(I),I=1,NR)
  590 FORMAT (/,5X,'NUMBER OF NODAL POINTS ON EACH RADIAL LINE IS:',/,
     *       16I5,/)
  600 WRITE(6,610)
  610 FORMAT(/,5X,'RADIAL DISTANCES OF THE NODES FROM THE ORIGIN ON EACH
     * RADIAL LINE:',/)
      DO 630 KK=1,NR
      IF (NTB .EQ. 0) ND(KK)=NP
      K=ND(KK)
      READ (5,570) (R(KK,J),J=1,K)
      WRITE (6,620) KK
  620 FORMAT(5X,'RADIAL DISTANCES ON RADIAL LINE (',I2,') :')
  630 WRITE(6,640) (R(KK,J),J=1,K)
  640 FORMAT(9X,8F10.5)
      WRITE(6,650)
  650 FORMAT(/,5X,'ANGLES BETWEEN RADIAL LINES AND FIRST RADIAL LINE:')
      READ (5,570) (O(I),I=1,NR)
      WRITE(6,640) (O(I),I=1,NR)
C
C***********************************************************************C
C      CALCULATE                                                         C
C        1. TOTAL NUMBER OF NODAL POINTS                                 C
C        2. ELEMENT NOS. WITH  PERMEABILITIES DIFFERENT FROM THOSE OF THE C
C           STANDARD ELEMENT OR SPECIFIC STORAGES DIFFERENT FROM ONE      C
```

```
C       3. OVERALL NODAL NUMBERS                                               C
C*****************************************************************************C
C
      NBB=NB
      IF (NB .EQ. 0) NBB=1
      MB=MAXO(JD,KP,KS,IE,KD,KH)
      IF (MB .EQ. 0) MB=1
      CALL NODP (NP,NR,NZ,NB,NM,KP,KS,NFG,NELEM,R,O,ND,NS,NND,NNP,NNR,
     *   NNZ,NO,MO,NQ,NH,MB,NBB,IE,IS,IH)
C
C*****************************************************************************C
C     READ IN VALUES OF PERMEABILITIES DIFFERENT FROM THOSE OF THE STANDARD   C
C        ELEMENT OR SPECIFIC STORAGE DIFFERENT FROM ONE                        C
C*****************************************************************************C
C
      DO 670 I=1,NELEM
      DO 660 J=1,3
  660 PERM(I,J)=SK(J)
  670 STOR(I)=1.
      IF (NM .EQ. 0) GO TO 750
      WRITE (6,680)
  680 FORMAT(/,5X,'PERMEABILITIES AND SPECIFIC STORAGES OF THE NON-STAND
     *ARD ELEMENTS :')
      WRITE (6,690)
  690 FORMAT (/,5X,'NNP',5X,'NNR',5X,'NNZ',5X,'ELEM. NO.',18X,
     *   'PERMEABILITY OR SPECIFIC STORAGE',/)
      DO 740 I=2,3
      IF (I .EQ. 2 .AND. KP .EQ. 0) GO TO 740
      IF (I .EQ. 3 .AND. KS .EQ. 0) GO TO 740
      KPP=KP
      IF (I .EQ. 3) KPP=KS
      DO 730 J=1,KPP
      IF (I .EQ. 3) GO TO 710
      LK=NO(J)
      READ (5,570) (PERM(LK,II),II=1,3)
```

```
      WRITE(6,700) NNP(J,I),NNR(J,I),NNZ(J,I),LK,(PERM(LK,II),II=1,3)
  700 FORMAT(3(3X,I5),6X,I5,8X,'PERM',5X,'XK=',F10.5,2X,'YK=',F10.5,2X,
     *       'ZK=',F10.5)
      GO TO 730
  710 LK=NH(J)
      READ (5,570) STOR(LK)
      WRITE(6,720) NNP(J,I),NNR(J,I),NNZ(J,I),LK,STOR(LK)
  720 FORMAT(3(3X,I5),6X,I5,8X,'STOR',23X,F10.5)
  730 CONTINUE
  740 CONTINUE
C
C***********************************************************************C
C     DETERMINE NODAL NUMBERS AT WHICH DRAWDOWNS ARE TO BE PRINTED OUT  C
C***********************************************************************C
C
  750 IF (JD .EQ. 0) GO TO 770
      DO 760 I=1,JD
  760 NWD(I)=(MO(NNP(I,1),NNR(I,1))-1)*NZ+NNZ(1,1)
C
C***********************************************************************C
C     CALCULATE THE NODAL NUMBERS AT WHICH DISCHARGES OR ZERO DRAWDOWNS C
C        ARE GIVEN                                                      C
C***********************************************************************C
C
  770 IF (KD .EQ. 0) GO TO 790
      DO 780 I=1,KD
  780 NQ(1)=(MO(NNP(I,5),NNR(I,5))-1)*NZ+NNZ(I,5)
  790 IF (KH .EQ. 0) GO TO 810
      DO 800 I=1,KH
  800 NH(I)=(MO(NNP(I,6),NNR(I,6))-1)*NZ+NNZ(1,6)
C
C***********************************************************************C
C     QT - THE QUANTITY OF DISCHARGE, EQUALS ONE WHEN ENTIRE AQUIFER    C
C          CONSIDERED                                                   C
C***********************************************************************C
```

```
C
  810 QT=1.
      IF (NFG .GE. 1) QT=6.28318/O(NR)
      WRITE (6,820) NO,NELEM
  820 FORMAT (/,5X,'TOTAL NUMBER OF NODES=',17X,I5,/,5X,
     *   'TOTAL NUMBER OF ELEMENTS=',14X,I5)
C
C***********************************************************************C
C     DETERMINE PLANE EQUATIONS AND COORDINATES OF NODAL POINTS        C
C***********************************************************************C
C
      CALL PLANE (LOP,ZC,R,O,ND,NP,NR,NZ,DV1,NO,MB,XZ)
C
C***********************************************************************C
C     CALCULATE NUMBER OF EQUATIONS,MAX. HALF BAND WIDTH, AND RANGE OF EACH C
C        PARTITION                                                     C
C***********************************************************************C
C
      CALL BAPTI (NO,NR,NZ,NB,NPART,NS,NEQ,NEQB,MA,NBB,NFG,NELEM,R,
     *   NF,NP,ND,MO,NSTART,NEND)
C
C***********************************************************************C
C     CALCULATE STIFFNESS MATRICES BY PARTITION AND STORE THEM INTO TO TAPE 8  C
C***********************************************************************C
C
      CALL STIFI(NP,NR,NZ,NB,LOP,NFG,NELEM,H,O,R,ND,NS,NND,STOR,KA,NBB)
      CALL TIMER (2,MT(1))
C
C***********************************************************************C
C     INITIALIZE THE DRAWDOWN ARRAY TO ZERO                            C
C***********************************************************************C
C
      DO 830 I=1,NEQ
      DO 830 J=1,NCOLN
  830 S(I,J)=0.
```

```
C
C*************************************************************************C
C     SOLVE DRAWDOWN FOR EACH NODE BY TIME INCREMENTS                     C
C*************************************************************************C
C
      MI=MA+NEQB-1
      IF (NFW .EQ. 0) GO TO 840
      MC=NF*NZ+NDP
      GO TO 850
  840 MC=NDP
  850 J=0
      IF (NP .LT. 3) GO TO 870
      DO 860 I=1,NR
      IF (R(I,3) .EQ. 0.) GO TO 860
      J=J+1
  860 CONTINUE
  870 MAA=NZ*(1+NF+J*NFW)
      NT=NT-1
      DO 1100 ITS=1,NT
      DT=T(ITS+1)-T(ITS)
      NC=NCOLN
      NAV=NEQB*(MA+NC)
      IF (S(1,1) .GT. 100.) GO TO 1110
C
C*************************************************************************C
C     FORM AND SOLVE SIMULTANEOUS EQUATIONS                               C
C*************************************************************************C
C
      CALL FASEI (DT,NZ,NR,KH,KD,NF,MB,NO,MA,NPA,ITS,NEQ,NFW,NDP,NEQB,
     *     NPART,NELEM,NCOLN,R,C,H,F,G,NQ,FQ,FT,MAXA,NSTART,NEND,NP,NH,
     *     EF,NAV,MI,MC,NC,MAA)
C
C*************************************************************************C
C     CALCULATE DIMENSIONAL OR DIMENSIONLESS DRAWDOWN FOR EACH NODE, AND  C
C        THEN PRINT OUT                                                   C
```

```
C*******************************************************************************C
C
      DO 890 I=1,NO
      S(I,2)=S(I,1)/QT
      IF (NREAL .LT. 1) GO TO 880
      S(I,2)=(BQ*S(I,2))/(BK*BL)
      GO TO 890
  880 S(I,2)=S(I,2)*DV1
  890 CONTINUE
      CALL TIMER (2,MT(2))
      IF (KH .EQ. 0) GO TO 910
      DO 900 I=1,KH
      S(NH(I),1)=0.
  900 S(NH(I),2)=S(NH(I),1)
  910 TT=T(ITS+1)
      IF (NREAL .EQ. 0) GO TO 920
      TT=TT*BS*BL*BL/BK
  920 IF (NREAL .GT. 0) GO TO 940
      WRITE(6,930)
  930 FORMAT (//,27X,'DIMENSIONLESS VALUES',/)
      GO TO 970
  940 WRITE (6,950)
  950 FORMAT (//,32X,'REAL VALUES',//)
      WRITE (6,960) T(ITS+1),TT
  960 FORMAT(10X,'DIMENSIONLESS TIME=',2X,F13.5,4X,'OR',4X,
     *   'REAL TIME=',2X,F13.5,/)
       GO TO 990
C
  970 WRITE (6,980) TT
  980 FORMAT (/,32X,'TIME=',2X,F13.5,/)
  990 IF (JD .EQ. 0) GO TO 1020
      WRITE (6,1000)
 1000 FORMAT (2(5X,'NOD',4X,'NNP',4X,'NNR',4X,'NNZ',7X,'DRAWDOWN',3X),/)
      WRITE (6,1010) (NWD(I),NNP(I,1),NNR(I,1),NNZ(I,1),S(NWD(I),2),I=1,
     * JD)
```

```
 1010 FORMAT (2(3X,I5,3(2X,I5),2X,E16.8))
      GO TO 1050
 1020 WRITE (6,1030)
 1030 FORMAT(5X,'NOD',7X,'DRAWDOWN',2(8X,'NOD',7X,'DRAWDOWN'),/)
      WRITE (6,1040) (I,S(I,2),I=1,ND)
 1040 FORMAT (3(3X,I5,2X,F16.8))
 1050 CONTINUE
      IF (MD .EQ. 0) GO TO 1080
      JK=0
 1060 IF (JK .GT. MPT) GO TO 1080
      IF (TP(JK) .EQ. T(ITS+1)) GO TO 1070
      JK=JK+1
      GO TO 1060
C
C***********************************************************************C
C     PLOT SHAPE OF THE AQUIFER,PLANE VIEW OF EACH ELEMENT AND ITS AVERAGE   C
C     PIEZOMETRIC HEAD IN A SPECIFIED TIME PERIOD FOR THE ASSIGNED PLANES    C
C***********************************************************************C
C
 1070 IV=NREAL
      CALL PHPLOT (NP,NR,NZ,NB,IE,RR,MD,TT,NFG,R,D,ND,MO,ME,NS,AN,
     *   NND,NNP,NNR,NO,NELEM,IH,XM,YM,MG,MB,NBB,IS,IV)
 1080 CONTINUE
      CALL TIMER (2,MT(3))
      MT(1)=MT(1)+MT(2)+MT(3)
      WRITE (6,1090) ITS,MT(2),MT(1)
 1090 FORMAT(/,5X,'TIME PERIOD=',2X,I3,2X,'TIME FOR EQ. SOLVER=',2X,I8,
     *   2X,'TOTAL TIME SPENT=',2X,I8)
 1100 CONTINUE
C
C***********************************************************************C
C     TERMINATE THE PLOT                                                  C
C***********************************************************************C
C
      CALL PLOT (0.0,0.0,999)
```

```
      CALL EXIT
      GO TO 1130
 1110 WRITE (6,1120) S(1,1)
 1120 FORMAT(/,5X,'ERROR MESSAGE :',/,7X,'DIMENSIONLESS DRAWDOWN AT THE
     *WELL IS',2X,E16.8,2X,'IS NOT REASONABLE ** STOP **')
 1130 CONTINUE
      STOP
      END
```

```
C
C*****************************************************************************C
C     SUBROUTINE FOR DETERMINING NODAL POINT NUMBERS, TOTAL NUMBER OF NODAL    C
C        POINTS, AND ELEMENT NUMBERS WITH PERMEABILITIES DIFFERENT FROM THOSE  C
C        OF THE STANDARD ELEMENT OR SPECIFIC STORAGES DIFFERENT FROM ONE       C
C*****************************************************************************C
C
      SUBROUTINE NODP (NP,NR,NZ,NB,NM,KP,KS,NFG,NELEM,R,O,ND,NS,NND,NNP,
     *   NNR,NNZ,NO,MO,NQ,NH,MB,NBB,IE,IS,IH)
      DIMENSION R(NR,NP),O(NR),ND(NR),NS(NBB),NND(NR),NNP(MB,6),NQ(MB)
      DIMENSION NNR(MB,6),NNZ(MB,6),MO(NP,NR),NH(MB),IS(MB)
      DIMENSION PERM(100,3),NOD(100,8),S(200,2),X(200,3),Z(200,3)
      COMMON PERM,NOD,S,X,Z
C
C*****************************************************************************C
C     DEFINE THE NUMBER OF EACH NODAL POINT ON THE TOP PLANE                   C
C*****************************************************************************C
C
      DO 10 I=1,NR
   10 MO(1,I)=1
      K=1
      DO 20 IR=2,NP
      DO 20 IO=1,NR
      IF (ND(IO) .LT. IR) GO TO 20
      IF (R(IO,IR) .EQ. 0.) GO TO 20
      K=K+1
      MO(IR,IO)=K
   20 CONTINUE
      IH=K
      NO=K*NZ
C
      LK=0
      N=0
      KPR=1
      KPS=1
```

```
      DO 470 KR=2,3
      IF (NP .LT. KR) GO TO 470
      DO 460 IR=KR,NP
      IF (KR .EQ. 2 .AND. IR .GT. 2) GO TO 460
      DO 450 IO=1,NR
      KJ=0
      IF (NFG .GT. 0 .AND. IO .EQ. NR) GO TO 40
      IF (NND(IO) .GE. (IR-1)) GO TO 40
      IF (IE .EQ. 0 .OR. NB .EQ. 0) GO TO 450
      KJ=1
   30 IF (KJ .GT. IE) GO TO 450
      IF(IR.EQ.NNP(KJ,4).AND.IO.EQ.NNR(KJ,4).AND.IS(KJ).EQ.3) GO TO 40
      KJ=KJ+1
      GO TO 30
   40 IF (NB .GT. 0) GO TO 50
      KLT=IO+1
      GO TO 100
   50 IF(ND(IO) .GE. IR .AND. R(IO,IR) .EQ. 0.) GO TO 450
      K=1
      IF (KR .EQ. 2) GO TO 90
   60 IF (IR-NS(K)) 90,80,70
   70 IF (K .GT. NB) K=NB
      IF (K .EQ. NB) GO TO 80
      K=K+1
      GO TO 60
   80 KK=2**(NB-K)
      KLT=IO+KK
      GO TO 100
   90 KK=2**(NB-K+1)
      KLT=IO+KK
  100 IF (KLT .GT. NR .AND. NFG .GT. 0) GO TO 450
      L=0
      O(1)=0.
  110 CONTINUE
      IF (L .GT. 0 .AND. KJ .GT. 0) GO TO 450
```

```
      DO 440 IZ=2,NZ
      IF (KLT .GT. NR) O(1)=6.28319
      IF (KLT .GT. NR) KLT=I
      LK=LK+1
      DO 120 I=1,8
  120 NOD(LK,I)=0
      IF (NM .EQ. 0) GO TO 170
C
C***********************************************************************C
C     CALCULATE THE ELEMENT NUMBERS WITH PERMEABILITIES DIFFERENT FROM   C
C        THOSE OF THE STANDARD ELEMENT OR SPECIFIC STORAGES DIFFERENT FROM C
C        ONE                                                           C
C***********************************************************************C
C
      DO 160 I=2,3
      IF (I .EQ. 2 .AND. KP .EQ. 0) GO TO 160
      IF (I .EQ. 3 .AND. KS .EQ. 0) GO TO 160
      IL=0
      KPI=KPR
      KPP=KP
      IF (I .EQ. 3) KPI=KPS
      IF (I .EQ. 3) KPP=KS
      IF (I .EQ. 2 .AND. KPI .GT. KP) GO TO 160
      IF (I .EQ. 3 .AND. KPI .GT. KS) GO TO 160
      DO 150 J=KPI,KPP
      IF(IR .NE. NNP(J,I) .OR. IO .NE. NNR(J,I) .OR. IZ .NE. NNZ(J,I)+1)
     *    GO TO 150
      IF (IL .GE. 1) GO TO 150
      IF (I .EQ. 3) GO TO 130
      KPR=J+1
      NQ(J)=LK
      GO TO 140
  130 KPS=J+1
      NH(J)=LK
  140 IL=IL+1
```

```
  150 CONTINUE
  160 CONTINUE
C
C*************************************************************************C
C      RELATE THE NODAL NUMBER IN EACH ELEMENT TO THE OVERALL NODAL NUMBER    C
C*************************************************************************C
C
  170 IF (KR .EQ. 3) GO TO 190
      NN=6
      NOD(LK,1)=IZ-1
      NOD(LK,2)=NOD(LK,1)+1
      NOD(LK,3)=(MO(IR,IO)-1)*NZ+IZ-1
      NOD(LK,4)=NOD(LK,3)+1
      NOD(LK,5)=(MO(IR,KLT)-1)*NZ+IZ-1
      NOD(LK,6)=NOD(LK,5)+1
  180 IF (KLT .GT. 1) GO TO 440
      NDI=NOD(LK,3)
      NDII=NOD(LK,4)
      NOD(LK,3)=NOD(LK,5)
      NOD(LK,4)=NOD(LK,6)
      NOD(LK,5)=NDI
      NOD(LK,6)=NDII
      GO TO 440
  190 JK=0
      IF (KJ .GT. 0) GO TO 220
      IF (ND(IO) -IR) 200,220,220
  200 IF(ND(KLT)-IR) 440,210,210
  210 NN=6
      NOD(LK,5)=(MO(IR,KLT)-1)*NZ+IZ-1
      NOD(LK,6)=NOD(LK,5)+1
      GO TO 310
  220 IF (ND(KLT)-IR) 230,250,250
  230 IF (NB .EQ. 0) GO TO 240
      IF (IR .EQ. NS(K)) GO TO 290
  240 NN=6
```

```
      GO TO 310
  250 IF (IE .EQ. 0) GO TO 280
      IF (KJ .GT. 0) GO TO 280
      JK=1
  260 IF (JK .GT. IE) GO TO 280
      IF(IR.EQ. NNP(JK,4).AND.IO.EQ.NNR(JK,4).AND.IZ.EQ.NNZ(JK,4)+1)
     *   GO TO 270
      JK=JK+1
      GO TO 260
  270 NN=6
      GO TO 310
  280 IF(NB .EQ. 0) GO TO 300
      IF (IR .NE. NS(K)) GO TO 300
      NN=6
      IF (R(KLT,IR-1) .EQ. 0.) GO TO 310
  290 IF (R(IO,IR-1) .EQ. 0. .AND. L .EQ. 1) GO TO 380
      NOD(LK,1)=(MO(IR-1,IO-KK)-1)*NZ+IZ-1
      GO TO 320
  300 NN=8
  310 NOD(LK,1)=(MO(IR-1,IO)-1)*NZ+IZ-1
  320 NOD(LK,2)=NOD(LK,1)+1
      IF (NB .EQ. 0) GO TO 330
      IF (IR .EQ. NS(K) .AND. R(KLT,IR-1) .EQ. 0.) GO TO 390
  330 NOD(LK,3)=(MO(IR-1,KLT)-1)*NZ+IZ-1
      NOD(LK,4)=NOD(LK,3)+1
      IF (IE .EQ. 0) GO TO 340
      IF (JK .EQ. 0 .OR. JK .GT. IE) GO TO 340
      IF (IS(JK) .EQ. 1) GO TO 350
      IF(IR.EQ.NNP(JK,4).AND.IO.EQ.NNR(JK,4).AND.IZ.EQ.NNZ(JK,4)+1)
     *     GO TO 400
  340 IF (KJ .GT. 0) GO TO 350
      IF (ND(IO) .LT. IR .AND. ND(KLT) .GE. IR) GO TO 410
  350 NOD(LK,5)=(MO(IR,IO)-1)*NZ+IZ-1
      NOD(LK,6)=NOD(LK,5)+1
      IF (IE .EQ. 0) GO TO 360
```

```
      IF (JK .EQ. 0 .OR. JK .GT. IE) GO TO 360
      IF (IS(JK) .EQ. 1) GO TO 410
  360 IF (KJ .GT. 0) GO TO 410
      IF (ND(IO) .GE. IR .AND. ND(KLT) .LT. IR) GO TO 410
      IF (NB .EQ. 0) GO TO 370
      IF (IR .EQ. NS(K) .AND. R(IO,IR-1) .EQ. 0) GO TO 410
  370 NOD(LK,7)=(MO(IR,KLT)-1)*NZ+IZ-1
      NOD(LK,8)=NOD(LK,7)+1
      IF (ND(IO) .GE. IR .AND. ND(KLT) .GE. IR) GO TO 410
      IF (NB .EQ. 0) GO TO 400
  380 IF (ND(KLT) .LT. IR) GO TO 440
      NOD(LK,1)=(MO(IR-1,KLT)-1)*NZ+IZ-1
      NOD(LK,2)=NOD(LK,1)+1
  390 NOD(LK,3)=(MO(IR,IO)-1)*NZ+IZ-1
      NOD(LK,4)=NOD(LK,3)+1
  400 NOD(LK,5)=(MO(IR,KLT)-1)*NZ+IZ-1
      NOD(LK,6)=NOD(LK,5)+1
      GO TO 180
  410 IF (KLT .GT. 1) GO TO 440
      NDI=NOD(LK,1)
      NDII=NOD(LK,2)
      NOD(LK,1)=NOD(LK,3)
      NOD(LK,2)=NOD(LK,4)
      NOD(LK,3)=NDI
      NOD(LK,4)=NDII
      IF (IE .EQ. 0) GO TO 420
      IF (JK .EQ. 0 .OR. JK .GT. IE) GO TO 420
      IF (IS(JK) .EQ. 1) GO TO 440
  420 IF (ND(IO) .LT. IR .AND. ND(KLT) .GE. IR) GO TO 440
      IF (ND(IO) .GE. IR .AND. ND(KLT) .LT. IR) GO TO 440
      IF (NB .EQ. 0) GO TO 430
      IF (IR .EQ. NS(K) .AND. R(IO,IR-1) .EQ. 0) GO TO 440
  430 NDI=NOD(LK,5)
      NDII=NOD(LK,6)
      NOD(LK,5)=NOD(LK,7)
```

```
      NOD(LK,6)=NOD(LK,8)
      NOD(LK,7)=NDI
      NOD(LK,8)=NDII
  440 CONTINUE
      IF (NB .EQ. 0) GO TO 450
      IF (IR .NE. NS(K)) GO TO 450
      IF (R(KLT,IR-1) .EQ. 0) GO TO 450
      IF (L .GE. 1) GO TO 450
      L=L+1
      GO TO 110
  450 CONTINUE
  460 CONTINUE
  470 CONTINUE
      NELEM=LK
      RETURN
      END
```

```
C
C***********************************************************************C
C     SUBROUTINE FOR DETERMINING PLANE EQUATIONS AND COORDINATES OF NODAL C
C        POINTS                                                          C
C***********************************************************************C
C
      SUBROUTINE PLANE (LOP,ZC,R,O,ND,NP,NR,NZ,DV1,NO,MB,XZ)
      DIMENSION ZC(2,3,3),A(2,3),B(2),ND(NR),R(NR,NP),O(NR),XZ(NZ)
      DIMENSION PERM(100,3),NOD(100,8),S(200,2),X(200,3),Z(200,3)
      COMMON PERM,NOD,S,X,Z
C
C***********************************************************************C
C     DETERMINE PLANE EQUATIONS                                          C
C***********************************************************************C
C
      IF (LOP .EQ. 1) GO TO 110
      DO 100 K=1,2
      B(K)=0.
      IF (LOP .EQ. 2 .AND. K .GT. 2) GO TO 60
      DO 50 I=1,3
      IF (I-2) 10,20,30
   10 J=2
      L=3
      GO TO 40
   20 J=3
      L=1
      GO TO 40
   30 J=1
      L=2
   40 A(K,I)=(ZC(K,1,J)-ZC(K,2,J))*(ZC(K,1,L)-ZC(K,3,L))-
     *    (ZC(K,1,J)-ZC(K,3,J))*(ZC(K,1,L)-ZC(K,2,L))
   50 CONTINUE
      IF (LOP .EQ. 3) GO TO 80
      IF (LOP .EQ. 2 .AND. K .EQ. 1) GO TO 80
   60 DO 70 I=1,3
```

```
   70 A(K,I)=A(K-1,I)
   80 DO 90 I=1,3
   90 B(K)=B(K)+A(K,I)*ZC(K,1,I)
  100 CONTINUE
C
C***********************************************************************C
C     DETERMINE THE COORDINATES OF THE NODAL POINTS                     C
C***********************************************************************C
C
  110 XNZ=NZ-1
      DV1=XZ(1)-XZ(NZ)
      DO 130 I=1,NZ
      DO 120 J=1,2
  120 Z(I,J)=0.
  130 Z(I,3)=XZ(I)
      L=NZ
      DO 210 I=2,NP
      DO 200 J=1,NR
      IF (L .EQ. NO) GO TO 200
      IF (ND(J) .LT. I) GO TO 200
      IF (R(J,I) .EQ. 0.) GO TO 200
      KK=L+1
      Z(KK,1)=R(J,I)*COS(O(J))
      Z(KK,2)=R(J,I)*SIN(O(J))
      IF (LOP .EQ. 1) GO TO 150
      VD1=(1./A(1,3))*(B(1)-A(1,1)*Z(KK,1)-A(1,2)*Z(KK,2))
      IF (LOP .EQ. 2) GO TO 140
      VD2=(1./A(2,3))*(B(2)-A(2,1)*Z(KK,1)-A(2,2)*Z(KK,2))
      DV=VD1-VD2
  140 Z(KK,3)=VD1
      IF (LOP .GT. 1) GO TO 160
  150 Z(KK,3)=XZ(1)
  160 DO 190 II=2,NZ
      KK=KK+1
      Z(KK,1)=Z(KK-1,1)
```

```
      Z(KK,2)=Z(KK-1,2)
      XII=II
      IF (LOP .LE. 2) GO TO 170
      Z(KK,3)=VD1-DV*(XZ(1)-XZ(II))/DV1
      IF (LOP .GT. 2) GO TO 190
  170 IF (LOP .LE. 1) GO TO 180
      Z(KK,3)=VD1-(XII-1)*DV1
      IF (LOP .GT. 1) GO TO 190
  180 Z(KK,3)=XZ(II)
  190 CONTINUE
      L=KK
  200 CONTINUE
  210 CONTINUE
      RETURN
      END
```

```
C*********************************************************************C
C     SUBROUTINE FOR CALCULATING NUMBER OF EQUATIONS,MAX. HALF BAND WIDTH,    C
C        AND RANGE OF EACH PARTITION                                          C
C*********************************************************************C
C
      SUBROUTINE BAPTI (NO,NR,NZ,NB,NPART,NS,NEQ,NEQB,MA,NBB,NFG,
     *   NELEM,R,NF,NP,ND,MO,NSTART,NEND)
      DIMENSION ND(NR),MO(NP,NR),NSTART(NPART),NEND(NPART),NS(NBB)
      DIMENSION PERM(100,3),NOD(100,8),S(200,2),X(200,3),Z(200,3)
      DIMENSION R(NR,NP)
      COMMON PERM,NOD,S,X,Z
C
C*********************************************************************C
C     CALCULATE TOTAL NUMBER OF EQUATIONS AND NUMBER OF EQUATIONS PER         C
C        PARTITION                                                            C
C*********************************************************************C
C
      NEQB=NO/NPART
      NEQ=NEQB*NPART
      IF (NEQ .EQ. NO) GO TO 10
      NEQB=(NO+NPART)/NPART
      NEQ=NEQB*NPART
C
C*********************************************************************C
C     DETERMINE MAXIMUM HALF BAND WIDTH                                       C
C*********************************************************************C
C
   10 IF (NB .GT. 0) GO TO 60
      IF (NFG .GT. 0) GO TO 20
      LK=(NZ-1)*(2*NR-1)+1
      GO TO 30
   20 LK=(NR-1)*(NZ-1)+1
   30 LN=8
      IF (NOD(LK,7) .EQ. 0) LN=6
      IF (LN .EQ. 8) GO TO 40
```

```
      I=MAXO(NOD(LK,1),NOD(LK,2),NOD(LK,3),NOD(LK,4),NOD(LK,5),NOD(LK,6)
     *)
      J=MINO(NOD(LK,1),NOD(LK,2),NOD(LK,3),NOD(LK,4),NOD(LK,5),NOD(LK,6)
     *)
      GO TO 50
   40 I=MAXO(NOD(LK,1),NOD(LK,2),NOD(LK,3),NOD(LK,4),NOD(LK,5),NOD(LK,6)
     *,NOD(LK,7),NOD(LK,8))
      J=MINO(NOD(LK,1),NOD(LK,2),NOD(LK,3),NOD(LK,4),NOD(LK,5),NOD(LK,6)
     *,NOD(LK,7),NOD(LK,8))
   50 MA=I-J+1
      NF=NR
      GO TO 110
   60 K=0
      DO 70 I=1,NR
      IF (R(I,2) .EQ. 0) GO TO 70
      K=K+1
   70 CONTINUE
      NF=K
      NK=NZ-1
      MA=0
      DO 100 I=1,NELEM,NK
      LN=8
      IF (NOD(I,7) .EQ. 0) LN=6
      IF (LN .EQ. 8) GO TO 80
      M=MAXO(NOD(I,1),NOD(I,2),NOD(I,3),NOD(I,4),NOD(I,5),NOD(I,6))
      L=MINO(NOD(I,1),NOD(I,2),NOD(I,3),NOD(I,4),NOD(I,5),NOD(I,6))
      GO TO 90
   80 M=MAXO(NOD(I,1),NOD(I,2),NOD(I,3),NOD(I,4),NOD(I,5),NOD(I,6),
     *   NOD(I,7),NOD(I,8))
      L=MINO(NOD(I,1),NOD(I,2),NOD(I,3),NOD(I,4),NOD(I,5),NOD(I,6),
     *   NOD(I,7),NOD(I,8))
   90 ML=M-L
      IF (MA .EQ. ML) GO TO 100
      MA=MAXO(ML,MA)
  100 CONTINUE
```

```
      MA=MA+1
  110 WRITE (6,120) NEQ,NEQB,MA
  120 FORMAT (5X,'TOTAL NUMBER OF EQUATIONS=',13X,I5,/,5X,
     *   'NUMBER OF EQUATIONS IN EACH PARTITION=',1X,I5,/,5X,
     *   'MAXIMUM HALF BAND WIDTH=',15X,I5)
C
C*****************************************************************************C
C     DETERMINE RANGE OF EACH PARTITION BY ELEMENT NUMBER                      C
C*****************************************************************************C
C
      WRITE (6,125)
  125 FORMAT(/,5X,'STARTING AND ENDING ELEMENTS IN EACH PARTITION ARE:',
     *   /)
      LKK=1
      DO 220 II=1,NPART
      IF (II .GT. 1) GO TO 130
      NSTART(II)=1
  130 LLK=LKK
      JJ=0
      ICC=0
      KN=II*NEQB
      KM=KN+MA-1
      KI=KN+1
      MK=(II-1)*NEQB+1
      IF (MK .GT. NO) GO TO 190
      MM=0
      DO 180 LK=LLK,NELEM
      LN=8
      IF (NOD(LK,7) .EQ. 0) LN=6
      IF (II .EQ. NPART) GO TO 150
      IF (ICC .EQ. 1) GO TO 150
      DO 140 JK=1,LN
      IF (ICC .EQ. 1) GO TO 140
      IF (NOD(LK,JK) .LT. KI) GO TO 140
      IF (NOD(LK,JK) .GT. (II+1)*NEQB) GO TO 140
```

```
      NSTART(II+1)=LK
      LKK=LK
      ICC=1
  140 CONTINUE
  150 ICI=0
      DO 170 I=1,LN
      IF (ICI .EQ. 1) GO TO 170
      IF(NOD(LK,I) .LT. MK) GO TO 170
      IF (NOD(LK,I) .GT. KN) GO TO 170
      DO 160 J=I,LN
      IF (ICI .EQ. 1) GO TO 160
      IF(NOD(LK,J) .GT. (NOD(LK,I)+MA-1)) GO TO 160
      MM=MAXO(MM,LK)
      ICI=1
  160 CONTINUE
  170 CONTINUE
  180 CONTINUE
      NEND(II)=MM
      GO TO 200
  190 NSTART(II)=0
      NEND(II)=0
  200 WRITE (6,210) II,NSTART(II),NEND(II)
  210 FORMAT (5X,'PARTITION NO.=',2X,I5,2X,'NSTART=',2X,I5,2X,
     *     'NEND=',2X,I5)
  220 CONTINUE
      RETURN
      END
```

```
C
C**************************************************************************C
C     SUBROUTINE FOR CHECKING THE SIMILARITY OF ELEMENTS, AND FOR CALCULATING C
C        AND STORING ALL STIFFNESS MATRICES BY ELEMENTS                    C
C**************************************************************************C
C
      SUBROUTINE STIFI (NP,NR,NZ,NB,LOP,NFG,NELEM,H,O,R,ND,NS,NND,STOR,
     *   KA,NBB)
      DIMENSION NS(NBB),NND(NR),STOR(NELEM)
      DIMENSION H(36,2),O(NR),R(NR,NP),ND(NR),KA(NELEM)
      DIMENSION PERM(100,3),NOD(100,8),S(200,2),X(200,3),Z(200,3)
      COMMON PERM,NOD,S,X,Z
      LK=0
      N=0
      DO 170 KR=2,3
      DO 170 IR=KR,NP
      IF (KR .EQ. 2 .AND. IR .GT. 2) GO TO 170
      DO 160 IO=1,NR
      IF (NFG .GT. 0 .AND. IO .EQ. NR) GO TO 10
      IF (NND(IO) .LT. (IR-1)) GO TO 160
   10 IF (NB .GT. 0) GO TO 20
      KLT=IO+1
      GO TO 70
   20 IF (R(IO,IR) .EQ. 0) GO TO 160
      K=1
      IF (KR .EQ. 2) GO TO 60
   30 IF (IR-NS(K)) 60,50,40
   40 IF (K .GT. NB) K=NB
      IF (K .EQ. NB) GO TO 50
      K=K+1
      GO TO 30
   50 KK=2**(NB-K)
      KLT=IO+KK
      GO TO 70
   60 KK=2**(NB-K+1)
```

```
      KLT=IO+KK
   70 IF (KLT .GT. NR .AND. NFG .GT. 0) GO TO 160
      L=0
      O(1)=0.
   80 CONTINUE
      DO 150 IZ=2,NZ
      IF (KLT .GT. NR) O(1)=6.28319
      IF (KLT .GT. NR) KLT=1
      LK=LK+1
      NN=8
      IF (NOD(LK,7) .EQ. 0) NN=6
C
C*****************************************************************************C
C     CHECK SIMILARITY OF ELEMENTS                                            C
C*****************************************************************************C
C
      IF (NB .NE. 0) GO TO 90
      KK=1
      GO TO 100
   90 IF (IR .EQ. NS(K) .AND. IZ .EQ. 2) GO TO 130
  100 IF (IO .EQ. 1 .AND. IZ .EQ. 2) GO TO 130
      IF (LOP .GE. 2) GO TO 130
      IF (ABS(PERM(LK,1)-PERM(LK-1,1)) .GT. 0.0001) GO TO 130
      IF (ABS(PERM(LK,2)-PERM(LK-1,2)) .GT. 0.0001) GO TO 130
      IF (ABS(PERM(LK,3)-PERM(LK-1,3)) .GT. 0.0001) GO TO 130
      IF (ABS(STOR(LK)-STOR(LK-1)) .GT. 0.0001) GO TO 130
      IF(IO .EQ. 1 .AND. IZ .LE. NZ) GO TO 140
      IF (IR .GT. ND(IO-KK)) R(IO-KK,IR)=0.
      IF (IR .GT. ND(IO)) R(IO,IR)=0.
      IF (IR .GT. ND(KLT)) R(KLT,IR)=0.
      IF (IZ .GT. 2) GO TO 140
      IF(NN .EQ. 8) GO TO 110
      IF (ABS((R(KLT,IR)-R(IO,IR))+(R(IO,IR)-R(IO-KK,IR))) .GT. 0.0001)
     *     GO TO 130
      IF (NN .EQ. 6) GO TO 120
```

```
  110 IF (ABS((R(KLT,IR)-R(KLT,IR-1))-(R(IO,IR)-R(IO,IR-1))-(R(IO,IR)-
     *  R(IO,IR-1))+(R(IO-KK,IR)-R(IO-KK,IR-1))) .GT. 0.0001) GO TO 130
  120 IF (ABS(O(IO)-O(IO-KK)-O(KLT)+O(IO)) .LE. 0.0001) GO TO 140
  130 N=N+1
  140 KA(LK)=N
  150 CONTINUE
      IF (NB .EQ. 0) GO TO 160
      IF (IR .NE. NS(K)) GO TO 160
      IF (R(KLT,IR-1) .EQ. 0) GO TO 160
      IF (L .GE. 1) GO TO 160
      L=L+1
      GO TO 80
  160 CONTINUE
  170 CONTINUE
C
C**********************************************************************C
C     CALCULATE STIFFNESS MATRICES BY ELEMENTS AND WRITE INTO TAPE 1   C
C**********************************************************************C
C
      DO 200 KL=1,NELEM
      IF (KL .EQ. 1) GO TO 180
      IF (KA(KL) .EQ. KA(KL-1)) GO TO 190
  180 NN=8
      IF (NOD(KL,7) .EQ. 0) NN=6
      K=NELEM
C
      CALL FEM(KL,H,NN,STOR,K)
C
  190 WRITE (8'KL) ((H(I,J),I=1,36),J=1,2)
  200 CONTINUE
      RETURN
      END
```

```
C
C*************************************************************************C
C      SUBROUTINE FOR CALCULATING STIFFNESS MATRICES OF ELEMENTS WITH SIX   C
C       OR EIGHT POINTS AND SUBDIVIDING THE ELEMENT WITH SIX NODAL POINTS   C
C       INTO SIX TETRAHEDRA AND THE ONE WITH EIGHT NODAL POINTS INTO TEN    C
C       TETRAHEDRA                                                          C
C*************************************************************************C
C
      SUBROUTINE FEM(KK,H,NN,STOR,K)
      DIMENSION TET(10,4),HH(10,4,4,2),H(36,2),STOR(K)
      DIMENSION PERM(100,3),NOD(100,8),S(200,2),X(200,3),Z(200,3)
      COMMON PERM,NOD,S,X,Z
      INTEGER TET
      IF (NN .EQ. 6) GO TO 40
      IT=10
      DO 10 I=1,4
   10 TET(I,1)=NOD(KK,1)
      DO 20 I=1,3
      TET(I,4)=NOD(KK,7)
      TET(5,I+1)=NOD(KK,I+5)
      TET(8,I)=NOD(KK,I)
      TET(10,I)=NOD(KK,I+1)
   20 TET(I+5,I+1)=NOD(KK,5)
      DO 30 I=1,2
      TET(1,I+1)=NOD(KK,I+2)
      TET(2,I+1)=NOD(KK,I+4)
      TET(6,I+2)=NOD(KK,I+6)
      TET(7,I)=NOD(KK,I+1)
      TET(9,I+1)=NOD(KK,I+4)
      TET(I+8,4)=NOD(KK,8)
      TET(I+2,I+2)=NOD(KK,6)
      TET(I+2,I+1)=NOD(KK,4)
   30 TET(I+4,1)=NOD(KK,5-I)
      TET(4,2)=NOD(KK,2)
      TET(7,4)=NOD(KK,8)
```

```
      TET(9,1)=NOD(KK,2)
      GO TO 80
   40 IT=6
      TET(1,4)=NOD(KK,5)
      DO 50 I=1,2
      TET(I,2)=NOD(KK,2)
      TET(I,3)=NOD(KK,I+2)
      TET(I+2,2)=NOD(KK,3)
      TET(I+2,3)=NOD(KK,I+3)
   50 TET(I+4,1)=NOD(KK,2)
      DO 60 I=1,3
      TET(I+1,4)=NOD(KK,6)
      TET(5,I+1)=NOD(KK,I+2)
   60 TET(6,I+1)=NOD(KK,I+3)
      DO 70 I=1,4
   70 TET(I,1)=NOD(KK,1)
   80 CONTINUE
      DO 90 I=1,36
      DO 90 L=1,2
   90 H(I,L)=0.
      DO 110 N=1,IT
      DO 100 I=1,4
      DO 100 J=1,4
      DO 100 LL=1,2
  100 HH(N,I,J,LL)=0.
  110 CALL TETRA (N,TET,HH,KK,STOR,K)
      L=1
      IF(IT-7) 130,130,120
  120 H( 1,L) =  (HH(1,1,1,L)+HH(2,1,1,L)+HH(3,1,1,L)+HH(4,1,1,L)+
     *HH(8,1,1,L))/2
      H( 2,L) =  (HH(4,1,2,L)+HH(8,1,2,L))/2
      H( 4,L) =  (HH(1,1,2,L)+HH(8,1,3,L))/2
      H( 7,L) =  (HH(1,1,3,L)+HH(3,1,2,L)+HH(4,1,3,L))/2
      H(11,L) =  (HH(2,1,2,L)+HH(8,1,4,L))/2
      H(16,L) =  (HH(2,1,3,L)+HH(3,1,3,L)+HH(4,1,4,L))/2
```

```
      H(22,L) =  (HH(1,1,4,L)+HH(2,1,4,L)+HH(3,1,4,L))/2
      H( 3,L) =  (HH(4,2,2,L)+HH(7,1,1,L)+HH(8,2,2,L)+HH(9,1,1,L)+
     *HH(10,1,1,L))/2
      H( 5,L) =  (HH(7,1,2,L)+HH(8,2,3,L)+HH(10,1,2,L))/2
      H( 8,L) =  (HH(4,2,3,L)+HH(10,1,3,L))/2
      H(12,L) =  (HH(7,1,3,L)+HH(8,2,4,L)+HH(9,1,2,L))/2
      H(17,L) =  (HH(4,2,4,L)+HH(9,1,3,L))/2
      H(30,L) =  (HH(7,1,4,L)+HH(9,1,4,L)+HH(10,1,4,L))/2
      H( 6,L) =  (HH(1,2,2,L)+HH(6,1,1,L)+HH(7,2,2,L)+HH(8,3,3,L)+
     *HH(10,2,2,L))/2
      H( 9,L) =  (HH(1,2,3,L)+HH(10,2,3,L))/2
      H(13,L) =  (HH(6,1,2,L)+HH(7,2,3,L)+HH(8,3,4,L))/2
      H(24,L) =  (HH(1,2,4,L)+HH(6,1,3,L))/2
      H(31,L) =  (HH(6,1,4,L)+HH(7,2,4,L)+HH(10,2,4,L))/2
      H(10,L) =  (HH(1,3,3,L)+HH(3,2,2,L)+HH(4,3,3,L)+HH(5,1,1,L)+
     *HH(10,3,3,L))/2
      H(19,L) =  (HH(3,2,3,L)+HH(4,3,4,L)+HH(5,1,2,L))/2
      H(25,L) =  (HH(1,3,4,L)+HH(3,2,4,L)+HH(5,1,3,L))/2
      H(32,L) =  (HH(5,1,4,L)+HH(10,3,4,L))/2
      H(15,L) =  (HH(2,2,2,L)+HH(6,2,2,L)+HH(7,3,3,L)+HH(8,4,4,L)+
     *HH(9,2,2,L))/2
      H(20,L) =  (HH(2,2,3,L)+HH(9,2,3,L))/2
      H(26,L) =  (HH(2,2,4,L)+HH(6,2,3,L))/2
      H(33,L) =  (HH(6,2,4,L)+HH(7,3,4,L)+HH(9,2,4,L))/2
      H(21,L) =  (HH(2,3,3,L)+HH(3,3,3,L)+HH(4,4,4,L)+HH(5,2,2,L)+
     *HH(9,3,3,L))/2
      H(27,L) =  (HH(2,3,4,L)+HH(3,3,4,L)+HH(5,2,3,L))/2
      H(34,L) =  (HH(5,2,4,L)+HH(9,3,4,L))/2
      H(28,L) =  (HH(1,4,4,L)+HH(2,4,4,L)+HH(3,4,4,L)+HH(5,3,3,L)+
     *HH(6,3,3,L))/2
      H(35,L) =  (HH(5,3,4,L)+HH(6,3,4,L))/2
      H(36,L) =  (HH(5,4,4,L)+HH(6,4,4,L)+HH(7,4,4,L)+HH(9,4,4,L)+
     *HH(10,4,4,L))/2
      IF(L .EQ. 2) GO TO 140
      L=2
```

```
      GO TO 120
  130 H( 1,L) =  (HH(1,1,1,L)+HH(2,1,1,L)+HH(3,1,1,L)+HH(4,1,1,L))/2
      H( 2,L) =  (HH(1,1,2,L)+HH(2,1,2,L))/2.
      H( 4,L) =  (HH(1,1,3,L)+HH(3,1,2,L)+HH(4,1,2,L))/2.
      H( 7,L) =  (HH(2,1,3,L)+HH(3,1,3,L))/2.
      H(11,L) =  (HH(1,1,4,L)+HH(4,1,3,L))/2.
      H(16,L) =  (HH(2,1,4,L)+HH(3,1,4,L)+HH(4,1,4,L))/2.
      H( 3,L) =  (HH(1,2,2,L)+HH(2,2,2,L)+HH(5,1,1,L)+HH(6,1,1,L))/2.
      H( 5,L) =  (HH(1,2,3,L)+HH(5,1,2,L))/2.
      H( 8,L) =  (HH(2,2,3,L)+HH(5,1,3,L)+HH(6,1,2,L))/2.
      H(12,L) =  (HH(1,2,4,L)+HH(5,1,4,L)+HH(6,1,3,L))/2.
      H(17,L) =  (HH(2,2,4,L)+HH(6,1,4,L))/2.
      H( 6,L) =  (HH(1,3,3,L)+HH(3,2,2,L)+HH(4,2,2,L)+HH(5,2,2,L))/2.
      H( 9,L) =  (HH(3,2,3,L)+HH(5,2,3,L))/2.
      H(13,L) =  (HH(1,3,4,L)+HH(4,2,3,L)+HH(5,2,4,L))/2.
      H(18,L) =  (HH(3,2,4,L)+HH(4,2,4,L))/2.
      H(10,L) =  (HH(2,3,3,L)+HH(3,3,3,L)+HH(5,3,3,L)+HH(6,2,2,L))/2.
      H(14,L) =  (HH(5,3,4,L)+HH(6,2,3,L))/2.
      H(19,L) =  (HH(2,3,4,L)+HH(3,3,4,L)+HH(6,2,4,L))/2.
      H(15,L) =  (HH(1,4,4,L)+HH(4,3,3,L)+HH(5,4,4,L)+HH(6,3,3,L))/2
      H(20,L) =  (HH(4,3,4,L)+HH(6,3,4,L))/2.
      H(21,L) =  (HH(2,4,4,L)+HH(3,4,4,L)+HH(4,4,4,L)+HH(6,4,4,L))/2
      IF(L .EQ. 2) GO TO 140
      L=2
      GO TO 130
  140 CONTINUE
      RETURN
      END
```

```
C
C*********************************************************************************C
C      SUBROUTINE FOR CALCULATING STIFFNESS MATRICES OF THE TETRAHEDRAL          C
C       ELEMENTS                                                                 C
C*********************************************************************************C
C
      SUBROUTINE TETRA(N,TET,HH,KK,STOR,KI)
      DIMENSION TET(10,4),HH(10,4,4,2),B(4),C(4),D(4),A(4),XX(3)
      DIMENSION CEN(3),XY(4,4),Y(4),STOR(KI)
      DIMENSION PERM(100,3),NOD(100,8),S(200,2),X(200,3),Z(200,3)
      COMMON PERM,NOD,S,X,Z
      INTEGER TET
      DO 10 I=1,3
      CEN(I)=(Z(TET(N,1),I)+Z(TET(N,2),I)+Z(TET(N,3),I)+Z(TET(N,4),I))
     */4.
      DO 10 J=1,4
   10 X(TET(N,J),I)=Z(TET(N,J),I)-CEN(I)
      DO 80 I=1,4
      IF(I-2) 20,30,40
   20 J=2
      M=3
      P=4
      GO TO 70
   30 J=3
      M=4
      P=1
      GO TO 70
   40 IF(I-3) 50,50,60
   50 J=4
      M=1
      P=2
      GO TO 70
   60 J=1
      M=2
      P=3
```

```
   70 CONTINUE
      A(I)=(X(TET(N,J),1)*(X(TET(N,M),2)*X(TET(N,P),3)-X(TET(N,P),2)*X(
     *      TET(N,M),3))+X(TET(N,M),1)*(X(TET(N,P),2)*X(TET(N,J),3)-X(
     *      TET(N,J),2)*X(TET(N,P),3))+X(TET(N,P),1)*(X(TET(N,J),2)*X(
     *      TET(N,M),3)-X(TET(N,M),2)*X(TET(N,J),3)))*(-1)**(I+1)
      B(I)=(X(TET(N,J),2)*(X(TET(N,P),3)-X(TET(N,M),3))+X(TET(N,M),2)*(X
     *      (TET(N,J),3)-X(TET(N,P),3))+X(TET(N,P),2)*(X(TET(N,M),3)-X(
     *      TET(N,J),3)))*(-1)**(I+1)
      C(I)=(X(TET(N,J),1)*(X(TET(N,M),3)-X(TET(N,P),3))+X(TET(N,M),1)*(X
     *      (TET(N,P),3)-X(TET(N,J),3))+X(TET(N,P),1)*(X(TET(N,J),3)-X(
     *      TET(N,M),3)))*(-1)**(I+1)
      D(I)=(X(TET(N,J),1)*(X(TET(N,P),2)-X(TET(N,M),2))+X(TET(N,M),1)*(X
     *      (TET(N,J),2)-X(TET(N,P),2))+X(TET(N,P),1)*(X(TET(N,M),2)-X(
     *      TET(N,J),2)))*(-1)**(I+1)
   80 CONTINUE
      DO 90 I=1,4
      DO 90 J=1,4
      AIAJ=A(I)*A(J)
      BIBJ=B(I)*B(J)
      CICJ=C(I)*C(J)
      DIDJ=D(I)*D(J)
   90 HH(N,I,J,1)=(PERM(KK,1)*BIBJ+PERM(KK,2)*CICJ+PERM(KK,3)*DIDJ)/6.
      VOL6=0.
      DO 100 I=1,4
  100 VOL6=VOL6+A(I)
      VOL6=ABS(VOL6)
      DO 150 I=1,4
      DO 140 J=1,4
      HH(N,I,J,1)=HH(N,I,J,1)/(VOL6)
      IF (I-J) 120,110,120
  110 HH(N,I,J,2)=VOL6/60.
      GO TO 130
  120 HH(N,I,J,2)=VOL6/120.
  130 HH(N,I,J,2)=HH(N,I,J,2)*STOR(KK)
  140 CONTINUE
```

```
  150 CONTINUE
      RETURN
      END
```

```
C
C***********************************************************************C
C     SUBROUTINE FOR FORMATION OF MATRICES BY PARTITIONS AND FOR SOLUTIONS OF  C
C        SIMULTANEOUS EQUATIONS                                           C
C***********************************************************************C
C
      SUBROUTINE FASEI (DT,NZ,NR,KH,KD,NF,MB,NO,MA,NPA,ITS,NEQ,NFW,NDP,
     *   NEQB,NPART,NELEM,NCOLN,R,C,H,F,G,NQ,FQ,FT,MAXA,NSTART,NEND,NP,
     *   NH,EF,NAV,MI,MC,NC,MAA)
      DIMENSION R(NR,NP),C(NAV),G(NAV),MAXA(MI)
      DIMENSION FT(NO),EF(MC,MAA),NSTART(NPART),NEND(NPART)
      DIMENSION H(36,2),F(NEQB,2),NQ(MB),FQ(MB),NH(MB)
      DIMENSION PERM(100,3),NOD(100,8),S(200,2),X(200,3),Z(200,3)
      COMMON PERM,NOD,S,X,Z
      MAB=MA*NEQB
      MA1=MA-1
      DO 10 I=1,MC
      DO 10 J=1,MAA
   10 EF(I,J)=0.
      DO 20 I=1,NO
   20 FT(I)=0.
      IF (KD .EQ. 0) GO TO 40
      DO 30 I=1,KD
      J=NQ(I)
   30 FT(J)=FT(J)+2.*FQ(I)
C
   40 NAA=NPA*NEQB
      REWIND 1
      IF (ITS-1) 60,60,50
   50 IF(ABS(S(NAA,2)) .GT. 1.0E-05 .AND. NPA .LT. NPART) NPA=NPA+1
      IF(ABS(S(NAA,2)) .GT. 1.0E-04 .AND. NPA .LT. NPART) NPA=NPA+1
      IF(ABS(S(NAA,2)) .GT. 1.0E-03 .AND. NPA .LT. NPART) NPA=NPA+1
   60 DO 580 II=1,NPA
      NST=NSTART(II)
      NEN=NEND(II)
```

```
      NRT=(II-1)*NEQB
      NLT=II*NEQB
      IF (NST .NE. 0) GO TO 90
      DO 70 I=1,MAB
   70 C(I)=0.
      DO 80 I=1,NEQB
      C(I)=1.0E10
      DO 80 J=1,NC
   80 F(I,J)=0.
      IF (NST .EQ. 0) GO TO 570
   90 JJ=1
  100 CONTINUE
      DO 110 I=1,MAB
  110 C(I)=0.
C
C*****************************************************************************C
C     FORM MATRICES C AND D FOR THE SIMULTANEOUS EQUATIONS; D-MATRIX          C
C        IS USED FOR CALCULATING THE COLUMN MATRIX ON THE RIGHT SIDE OF       C
C        SIMULTANEOUS EQUATIONS                                               C
C*****************************************************************************C
C
      DO 160 LK=1,NELEM
      IF (LK .GT. NEN) GO TO 170
      IF (LK .LT. NST) GO TO 160
      NN=8
      IF (NOD(LK,7) .EQ. 0) NN=6
      JK=1
  120 IF(NOD(LK,JK) .GT. NRT .AND. NOD(LK,JK) .LE. NLT) GO TO 130
      IF (JK .EQ. NN) GO TO 160
      JK=JK+1
      GO TO 120
  130 READ (8'LK) ((H(I,J),I=1,36),J=1,2)
      DO 150 LL=1,NN
      IF (NOD(LK,LL) .GT. NLT) GO TO 150
      IF (NOD(LK,LL) .LE. NRT) GO TO 150
```

```
      DO 140 KK=LL,NN
      IF (NOD(LK,KK) .GT. (NOD(LK,LL)+MA-1)) GO TO 140
      LLK=(KK+1)*KK/2+LL-KK
      NDN=NOD(LK,LL)+NEQB*(NOD(LK,KK)-NOD(LK,LL))-(II-1)*NEQB
      C(NDN)=C(NDN)+H(LLK,1)*(-1)**JJ+2.*H(LLK,2)/DT
  140 CONTINUE
  150 CONTINUE
  160 CONTINUE
C
C***********************************************************************C
C     CALCULATE COLUMN MATRIX ON THE RIGHT SIDE OF SIMULTANEOUS EQUATIONS      C
C***********************************************************************C
C
  170 IF (JJ .EQ. 2) GO TO 430
C
C***********************************************************************C
C     MULTIPLY THE D MATRIX BY THE PREVIOUS DRAWDOWN                    C
C***********************************************************************C
C
      DO 220 I=1,NEQB
      IF (NCOLN .EQ. 2) F(I,2)=0.
      JK=(II-1)*NEQB+I
      IF (JK .GT. NO) GO TO 180
      F(I,1)=FT(JK)
      GO TO 190
  180 F(I,1)=0.
  190 IF (ITS .EQ. 1) GO TO 220
      DO 200 J=1,MA
      JK=(II-1)*NEQB+I+J-1
      IF (JK .GT. NEQ) GO TO 200
      JB=I+(J-1)*NEQB
      F(I,1)=F(I,1)+C(JB)*S(JK,1)
  200 CONTINUE
      DO 210 J=1,MA1
      IF (I .LE. J) GO TO 210
```

```
      JK=(II-1)*NEQB+I-J
      IF (JK .GT. NO) GO TO 210
      JB=I+J*(NEQB-1)
      F(I,1)=F(I,1)+C(JB)*S(JK,1)
  210 CONTINUE
  220 CONTINUE
C
C***********************************************************************C
C     CONSIDER BOUNDARY CONDITIONS AT THE WELL                         C
C***********************************************************************C
C
      JK=(II-1)*NEQB+1
      JU=II*NEQB
      IF (JK .GT. MC) GO TO 390
      DO 380 I=1,NDP
      IF (NCOLN .EQ. 1) GO TO 360
C
C***********************************************************************C
C     SET THE BOUNDARY CONDITIONS FOR NONUNIFORM DISCHARGE             C
C***********************************************************************C
C
      IF (I .LT. JK .OR. I .GT. JU) GO TO 230
      JW=I-JK+1
      F(JW,1)=(S(I,1)+1.)*1.0E10
      F(JW,2)=F(JW,1)
  230 IF (NFW .EQ. 0) GO TO 250
      JW=0
      DO 240 JVV=1,NR
      IF (R(JVV,2) .EQ. 0.) GO TO 240
      JW=JW+1
      KT=I+JW*NZ
      IF (KT .LT. JK .OR. KT .GT. JU) GO TO 240
      KTT=KT-JK+1
      F(KTT,1)=(S(KT,1)+1.)*1.0E10
      F(KTT,2)=F(KTT,1)
```

```
  240 CONTINUE
C
C*************************************************************************C
C      THE D-MATRIX FOR ONLY THOSE NODAL POINTS AT THE WELL ARE MULTIPLIED     C
C      BY THE PREVIOUS DRAWDOWN FOR THE CASE OF NONUNIFORM DISCHARGE AND THE   C
C      RESULTS ARE STORED TO BE USED LATER                                     C
C*************************************************************************C
C
  250 QN=0.
      KK=0
      NA=1
  260 DO 340 JV=1,NA
      KT=I+JV*NZ
      IF (KK .EQ. 0) KT=I
      IF (KT .LT. JK .OR. KT .GT. JU) GO TO 340
      KTT=KT-JK+1
      DO 270 J=1,MA
      JN=KT+J-1
      NJ=KTT+(J-1)*NEQB
      IF (JN .GT. MAA) GO TO 270
      QN=QN+C(NJ)*S(JN,1)
  270 CONTINUE
      DO 280 J=1,MA1
      IF (KTT .LE. J) GO TO 280
      JN=KT-J
      NJ=KTT+J*(NEQB-1)
      QN=QN+C(NJ)*S(JN,1)
      JN=KT-J
      NJ=KT+J*(NEQB-1)
  280 CONTINUE
      IF (NEQB .GE. MC) GO TO 340
      DO 330 J=1,MC
      JN=KT+J-1
      IF (JN .GT. MC) GO TO 330
      KN=JU-KT+2
```

```
      IF (J .LT. KN) GO TO 330
      IF (NDP .EQ. NZ) GO TO 320
      IA=NZ-NDP
      ID=0
      DO 310 IC=1,IA
      IB=NDP+IC-1
      IF (JN .NE. IB) GO TO 290
      ID=1
      GO TO 310
  290 JW=0
      DO 300 JVV=1,NR
      IF (R(JVV,2) .EQ. 0.) GO TO 300
      JW=JW+1
      NJ=IB+JW*NZ
      IF (JN .NE. NJ) GO TO 300
      ID=1
  300 CONTINUE
  310 CONTINUE
      IF (ID .EQ. 1) GO TO 330
  320 NJ=KTT+(J-1)*NEQB
      QN=QN+C(NJ)*S(KT,1)
  330 CONTINUE
  340 CONTINUE
      IF (NFW .EQ. 0) GO TO 380
      IF (KK .EQ. 1) GO TO 380
      KK=1
  350 NA=NF
      GO TO 260
C
C*****************************************************************************C
C     SET BOUNDARY CONDITIONS AT THE WELL FOR UNIFORM DISCHARGE               C
C*****************************************************************************C
C
  360 IF (I .LT. JK .OR. I .GT. JU) GO TO 380
      JW=I-JK+1
```

```
      HNZ=NDP
      IF (I .EQ. 1 .OR. I .EQ. NDP) GO TO 370
      F(JW,1)=F(JW,1)+2./(HNZ-1.)
      GO TO 380
  370 F(JW,1)=F(JW,1)+1./(HNZ-1.)
  380 CONTINUE
C
  390 IF (II .EQ. NPA) GO TO 420
C
C***********************************************************************C
C     MULTIPLY THE TRANSPOSE  OF D-MATRIX BY THE PREVIOUS DRAWDOWN FOR THE     C
C        LOWER BAND OF THE SUBSEQUENT PARTITIONS                               C
C***********************************************************************C
C
      DO 410 I=1,MA1
      JN=II*NEQB+I
      IF (JN .GT. NO) GO TO 410
      DO 400 J=1,MA1
      IF (I .GT. J .OR. (NEQB+I) .LE. J) GO TO 400
      JW=NEQB*(1+J)-J+I
      IF (JW .GT. MAB) GO TO 400
      KT=II*NEQB+I-J
      FT(JN)=FT(JN)+C(JW)*S(KT,1)
  400 CONTINUE
  410 CONTINUE
  420 JJ=2
      GO TO 100
C
C***********************************************************************C
C     DETERMINE THE BOUNDARY CONDITIONS AT THE WELL AND STORE THE C-           C
C     MATRIX AT THE WELL NODAL POINTS TO BE USED LATER FOR THE CALCULATION     C
C     OF THE CORRECTED DRAWDOWN IN THE CASE OF NONUNIFORM DISCHARGE            C
C***********************************************************************C
C
  430 JK=(II-1)*NEQB+1
```

```
      JU=II*NEQB
      IF (NCOLN .EQ. 1) GO TO 500
      IF (JK .GT. MC) GO TO 500
      IF (MC .GE. JU) GO TO 440
      NA=MC-JK+1
      GO TO 450
  440 NA=NEQB
  450 DO 460 JV=1,NA
      KT=JK+JV-1
      DO 460 J=1,MA
      JN=JV+(J-1)*NEQB
      NJ=KT+J-1
      IF (NJ .GT. MAA) GO TO 460
      EF(KT,NJ)=C(JN)
  460 CONTINUE
      DO 490 I=1,NDP
      IF (NFW .EQ. 0) GO TO 480
      DO 470 JV=1,NF
      KT=I+JV*NZ
      IF (KT .LT. JK .OR. KT .GT. JU) GO TO 470
      KT=KT-JK+1
      C(KT)=1.0E10
  470 CONTINUE
  480 IF (I .LT. JK .OR. I .GT. JU) GO TO 490
      KT=I-JK+1
      C(KT)=1.0E10
  490 CONTINUE
  500 IF (NEQ .EQ. NO) GO TO 550
      IF (JU .LE. NO) GO TO 550
      IF (JK-NO) 510,520,520
  510 KM=NO-JK+2
      GO TO 530
  520 KM=1
  530 DO 540 I=KM,NEQB
  540 C(I)=1.0E10
```

```
  550 IF (KH .EQ. 0) GO TO 570
      DO 560 I=1,KH
      IF (NH(I) .LT. JK .OR. NH(I) .GT. JU) GO TO 560
      KT=NH(I)-JK+1
      C(KT)=1.0E10
  560 CONTINUE
  570 WRITE(1) (C(I),I=1,MAB),((F(I,J),I=1,NEQB),J=1,NC)
  580 CONTINUE
C
C*****************************************************************************C
C     SOLVE SIMULTANEOUS EQUATIONS                                            C
C*****************************************************************************C
C
      NOQ=NEQB*NPA
      CALL SESOL (C,G,MAXA,NOQ,MA,NC,NPA,NEQB,NAV,MI,1,2,3,4)
C
      REWIND 3
      DO 590 II=1,NPA
      JJ=NPA-II
      M=JJ*NEQB+1
      N=(JJ+1)*NEQB
  590 READ(3) ((S(I,J),I=M,N),J=1,NC)
      IF (NCOLN .EQ. 1) GO TO 650
C
C*****************************************************************************C
C     1. CALCULATE QUANTITY OF FLOW FROM NODAL POINTS ALONG THE WELL BORE     C
C     2. MULTIPLY THE EF-MATRIX ALONG THE WELL BORE BY S(I,2)                 C
C*****************************************************************************C
C
      DO 600 I=1,MC
      DO 600 J=I,MC
  600 EF(J,I)=EF(I,J)
      Q=0.
      Q=Q-(QN/2.)
      ES=0.
```

```
      DO 630 I=1,NDP
      DO 630 J=1,MAA
      IF (J .GT. NO) GO TO 630
      IF (NFW .EQ. 0) GO TO 620
      JV=0
      DO 610 JVV=1,NR
      IF (R(JVV,2) .EQ. 0.) GO TO 610
      JV=JV+1
      KT=I+JV*NZ
      Q=Q+(EF(KT,J)*S(J,1))/2.
      ES=ES+EF(KT,J)*S(J,2)
  610 CONTINUE
  620 Q=Q+(EF(I,J)*S(J,1))/2.
      ES=ES+EF(I,J)*S(J,2)
  630 CONTINUE
C
C*****************************************************************************C
C     DEFINE THE CONSTANT 'ALPHA'                                             C
C*****************************************************************************C
C
      ALPHA=2.*(1-Q)/ES
C
C*****************************************************************************C
C     CALCULATE THE DRAWDOWN FOR A NONUNIFORMLY DISCHARGING WELL              C
C*****************************************************************************C
C
      DO 640 I=1,NO
      S(I,1)=S(I,1)+S(I,2)*ALPHA
  640 CONTINUE
  650 CONTINUE
      RETURN
      END
```

```
      C
      C*************************************************************************C
      C     SUBROUTINE FOR SOLVING SIMULTANEOUS EQUATIONS                       C
      C*************************************************************************C
      C
            SUBROUTINE SESOL (A,B,MAXA,NEQ,MA,NV,NBLOCK,NEQB,NAV,MI,NSTIF,
           *   NRED,NL,NR)
            DIMENSION A(NAV),B(NAV),MAXA(MI)
      C
            MM=1
            MA2=MA - 2
            IF (MA2.EQ.0) MA2=1
            INC=NEQB - 1
            NWA=NEQB*MA
            NTB=(MA-2)/NEQB + 1
            NEB=NTB*NEQB
            NEBT=NEB + NEQB
            NWV=NEQB*NV
            NWVV=NEBT*NV
      C
            N1=NL
            N2=NR
            REWIND NSTIF
            REWIND NRED
            REWIND N1
            REWIND N2
      C
      C*************************************************************************C
      C     START THE LOOP FROM FIRST TO LAST PARTITION                         C
      C*************************************************************************C
      C
            DO 410 NJ=1,NBLOCK
            IF (NJ.NE.1) GO TO 40
            READ (NSTIF) A
            IF (NEQ.GT.1) GO TO 50
```

```
      MAXA(1)=1
      WRITE (NRED) A,MAXA
      IF (A(1)) 10,120,20
   10 KK=1
      WRITE (6,600) KK,A(1)
   20 DO 30 L=1,NV
   30 A(1+L)=A(1+L)/A(1)
      KR=1 + NV
      WRITE (NL) (A(KK),KK=2,KR)
      RETURN
   40 IF (NTB.EQ.1) GO TO 50
      REWIND N1
      REWIND N2
      READ (N1) A
C
C*****************************************************************************C
C     FIND COLUMN HEIGHTS                                                      C
C*****************************************************************************C
C
   50 KU=1
      KM=MINO(MA,NEQB)
      MAXA(1)=1
      DO 110 N=2,MI
      IF (N-MA) 60,60,70
   60 KU=KU+ NEQB
      KK=KU
      MM=MINO(N,KM)
      GO TO 90
   70 KU=KU + 1
      KK=KU
      IF (N-NEQB) 90,90,80
   80 MM=MM -1
   90 DO 100 K=1,MM
      IF (A(KK)) 110,100,110
  100 KK=KK - INC
```

```
  110 MAXA(N)=KK
C
      IF (A(1)) 130,120,140
  120 KK=(NJ-1)*NEQB + 1
      IF (KK.GT.NEQ) GO TO 400
      WRITE (6,590) KK
      STOP
  130 KK=(NJ-1)*NEQB + 1
      WRITE (6,600) KK,A(1)
C
C**********************************************************************C
C     FACTORIZE LEADING PARTITION                                      C
C**********************************************************************C
C
  140 DO 250 N=2,NEQB
      NH=MAXA(N)
      IF (NH-N) 250,250,150
  150 KL=N + INC
      K=N
      D=0.
      DO 160 KK=KL,NH,INC
      K=K -1
      C=A(KK)/A(K)
      D=D + C*A(KK)
  160 A(KK)=C
      A(N)=A(N) -D
C
      IF (A(N)) 180,170,190
  170 KK=(NJ-1)*NEQB + N
      IF (KK.GT.NEQ) GO TO 400
      WRITE (6,590) KK
      STOP
  180 KK=(NJ-1)*NEQB + N
      WRITE (6,600) KK,A(N)
C
```

```
  190 IC=NEQB
      DO 220 J=1,MA2
      MJ=MAXA(N+J) -IC
      IF (MJ-N) 220,220,200
  200 KU=MINO(MJ,NH)
      KN=N + IC
      C=0.
      DO 210 KK=KL,KU,INC
  210 C=C + A(KK)*A(KK+IC)
      A(KN)=A(KN) - C
  220 IC=IC + NEQB
C
      K=N + NWA
      DO 240 L=1,NV
      KJ=K
      C=0.
      DO 230 KK=KL,NH,INC
      KJ=KJ -1
  230 C=C + A(KK)*A(KJ)
      A(K)=A(K) - C
  240 K=K + NEQB
C
  250 CONTINUE
C
C**************************************************************************C
C     CARRY OVER INTO TRAILING PARTITIONS                                  C
C**************************************************************************C
C
      DO 390 NK=1,NTB
      IF ((NK+NJ).GT.NBLOCK) GO TO 390
      NI=N1
      IF ((NJ.EQ.1).OR.(NK.EQ.NTB)) NI=NSTIF
      READ (NI) B
      ML=NK*NEQB + 1
      MR=MINO((NK+1)*NEQB,MI)
```

```
      IF (MA.EQ.1) ML=MR
      MD=MI - ML
      KL=NEQB + (NK-1)*NEQB*NEQB
      N=1
C
      DO 360 M=ML,MR
      NH=MAXA(M)
      KL=KL + NEQB
      IF (NH-KL) 350,260,260
  260 K=NEQB
      D=0.
      DO 270 KK=KL,NH,INC
      C=A(KK)/A(K)
      D=D + C*A(KK)
      A(KK)=C
  270 K=K - 1
      B(N)=B(N) - D
      IF (MD) 320,320,280
  280 IC = NEQB
      DO 310 J=1,MD
      MJ=MAXA(M+J) - IC
      IF (MJ-KL) 310,290,290
  290 KU=MINO(MJ,NH)
      KN=N+IC
      C=0.
      DO 300 KK=KL,KU,INC
  300 C=C + A(KK)*A(KK+IC)
      B(KN)=B(KN) - C
  310 IC=IC + NEQB
C
  320 KN=N + NWA
      K=NEQB + NWA
      DO 340 L=1,NV
      KJ=K
      C=0.
```

```
      DO 330 KK=KL,NH,INC
      C=C + A(KK)*A(KJ)
  330 KJ=KJ - 1
      B(KN)=B(KN) - C
      KN=KN + NEQB
  340 K=K + NEQB
C
  350 MD=MD - 1
  360 N=N + 1
C
      IF (NTB.NE.1) GO TO 380
      WRITE (NRED) A,MAXA
      DO 370 I=1,NAV
  370 A(I)=B(I)
      GO TO 410
  380 WRITE (N2) B
C
  390 CONTINUE
C
      M=N1
      N1=N2
      N2=M
  400 WRITE (NRED) A,MAXA
C
  410 CONTINUE
C
C**********************************************************************C
C     BACK SUBSTITUTE TO OBTAIN SOLUTION                               C
C**********************************************************************C
C
      DO 420 K=1,NWVV
  420 B(K)=0.
      REWIND NL
C
      DO 580 NJ=1,NBLOCK
```

```
      BACKSPACE NRED
      READ (NRED) A,MAXA
      BACKSPACE NRED
      K=NEBT
      DO 440 L=1,NV
      DO 430 I=1,NEB
      B(K)=B(K-NEQB)
  430 K=K - 1
  440 K=K + NEBT +NEB
      KN=0
      KK=NWA
      NDIF=NEQB
      IF (NJ.EQ.1) NDIF=NEQB - (NBLOCK*NEQB - NEQ)
      DO 460 L=1,NV
      DO 450 K=1,NDIF
  450 B(KN+K)=A(KK+K)/A(K)
      KK=KK + NEQB
  460 KN=KN + NEBT
      IF (MA.EQ.1) GO TO 550
      ML=NEQB + 1
      KL=NEQB
      DO 500 M=ML,MI
      KL=KL + NEQB
      KU=MAXA(M)
      IF (KU-KL) 500,470,470
  470 K=NEQB
      KM=M
      DO 490 L=1,NV
      KJ=K
      DO 480 KK=KL,KU,INC
      B(KJ)=B(KJ) - A(KK)*B(KM)
  480 KJ=KJ - 1
      KM=KM + NEBT
  490 K=K + NEBT
  500 CONTINUE
```

```
      N=NEQB
      DO 540 I=2,NEQB
      KL=N + INC
      KU=MAXA(N)
      IF (KU-KL) 540,510,510
  510 K=N
      DO 530 L=1,NV
      KJ=K
      DO 520 KK=KL,KU,INC
      KJ=KJ - 1
  520 B(KJ)=B(KJ) - A(KK)*B(K)
  530 K=K + NEBT
  540 N=N - 1
C
  550 KK=0
      KN=0
      DO 570 L=1,NV
      DO 560 K=1,NEQB
      KK=KK +1
  560 A(KK)=B(KN+K)
  570 KN=KN + NEBT
C
      WRITE (NL) (A(K),K=1,NWV)
  580 CONTINUE
C
  590 FORMAT (// 46H   STOP  ***  ZERO DIAGONAL ENCOUNTERED DURING,
     1           18H EQUATION SOLUTION, /
     2       13X,18H EQUATION NUMBER =, I6 )
  600 FORMAT (/  50H WARNING ***  NEGATIVE DIAGONAL ENCOUNTERED DURING,
     1           18H EQUATION SOLUTION, /
     2       13X,18H FOUATION NUMBER =, I6, 5X, 7HVALUE =, E20.8 )
C
      RETURN
      END
```

```
C
C****************************************************************************C
C      SUBROUTING FOR PLOTTING                                              C
C****************************************************************************C
C
       SUBROUTINE PHPLOT(NP,NR,NZ,NB,IE,RR,MD,TT,NFG,R,O,ND,MO,ME,NS,AN,
     *    NND,NNP,NNR,NO,NELEM,IH,XM,YM,MG,MB,NBB,IS,IV)
       DIMENSION AN(20),DATA(1024),XM(IH),YM(IH),ME(MD)
       DIMENSION O(NR),NS(NBB),ND(NR),NND(NR),IS(MB)
       DIMENSION R(NR,NP),MO(NP,NR),MG(NR,NP),NNP(MB,6),NNR(MB,6)
       DIMENSION PERM(100,3),NOD(100,8),S(200,2),X(200,3),Z(200,3)
       COMMON PERM,NOD,S,X,Z
C
C****************************************************************************C
C      EXECUTE PLOTTING FOR EACH PLANE                                      C
C****************************************************************************C
C
       XM(1)=0.
       YM(1)=XM(1)
       XM(2)=2.*RR
       YM(2)=XM(2)
       K=2
       CALL SCALE (XM,10.0,K,1)
       CALL SCALE (YM,10.0,K,1)
       XA=XM(3)
       YA=YM(3)
       XB=XM(4)
       YB=YM(4)
       XR=5.*XB
       YR=5.*YB
       DO 630 II=1,MD
       DO 30 I=1,NR
       K=ND(I)
       MH=0
       DO 20 J=1,K
```

```
      IF (J .GT. 1) GO TO 10
      IF (R(I,J+1) .EQ. 0.) GO TO 20
   10 IF (J .NE. 1 .AND. R(I,J) .EQ. 0.) GO TO 20
      MH=MH+1
      MG(I,J)=(MO(J,I)-1)*NZ+ME(II)
      XM(MH)=Z(MG(I,J),1)+XR
      YM(MH)=Z(MG(I,J),2)+YR
   20 CONTINUE
      K=MH
      XM(MH+1)=XA
      YM(MH+1)=YA
      XM(MH+2)=XB
      YM(MH+2)=YB
      CALL LINE (XM,YM,K,1,0,0)
   30 CONTINUE
      DO 270 I=2,NP
      MH=0
      MI=0
      KN=1
      IF (NB .GT. 0) GO TO 40
      KK=1
      GO TO 80
   40 IF (I-NS(KN)) 70,60,50
   50 IF(KN .GT. NB) K=NB
      IF (KN .EQ. NB) GO TO 60
      KN=KN+1
      GO TO 40
   60 KK=2**(NB-KN)
      GO TO 80
   70 KK=2**(NB-KN+1)
   80 MS=NR
      IF (NFG .EQ. 0) MS=NR+1
      O(1)=0.
      DO 260 J=1,MS,KK
      IF (O(1) .EQ. 6.28318) GO TO 260
```

```
      IF (J .LE. NR) GO TO 90
      IF (NND(J-KK) .LT. (I-1)) GO TO 260
      JK=1
      O(1)=6.28318
      GO TO 190
   90 IF (NFG .GT. 0 .AND. J .EQ. NR) GO TO 180
      IF (ND(J)-(I-1)) 260,100,100
  100 IF (NND(J) .LT. (I-1)) GO TO 170
      IF (ND(J)-I) 110,120,120
  110 MH=MH+1
      XM(MH)=Z(MG(J,I-1),1)+XR
      YM(MH)=Z(MG(J,I-1),2)+YR
      GO TO 260
  120 IF (IE .EQ. 0) GO TO 140
      K=0
  130 K=K+1
      IF(I .EQ. NNP(K,4) .AND. J .EQ. NNR(K,4)) GO TO 150
      IF (K .LT. IE) GO TO 130
  140 MH=MH+1
      XM(MH)=Z(MG(J,I),1)+XR
      YM(MH)=Z(MG(J,I),2)+YR
      GO TO 260
  150 MH=MH+1
      XM(MH)=Z(MG(J,I),1)+XR
      YM(MH)=Z(MG(J,I),2)+YR
      MH=MH+1
      IF (IS(K) .EQ. 2) GO TO 160
      JKK=J+KK
      IF (NFG .EQ. 0 .AND. JKK .GT. NR) JKK=1
      XM(MH)=Z(MG(JKK,I-1),1)+XR
      YM(MH)=Z(MG(JKK,I-1),2)+YR
      GO TO 260
  160 XM(MH)=Z(MG(J,I-1),1)+XR
      YM(MH)=Z(MG(J,I-1),2)+YR
      GO TO 260
```

```
170 IF (J .EQ. 1) GO TO 260
180 IF (NND(J-KK) .LT. (I-1)) GO TO 260
    JK=J
190 MH=MH+1
    IF (ND(JK)-I) 210,200,200
200 XM(MH)=Z(MG(JK,I),1)+XR
    YM(MH)=Z(MG(JK,I),2)+YR
    GO TO 220
210 XM(MH)=Z(MG(JK,I-1),1)+XR
    YM(MH)=Z(MG(JK,I-1),2)+YR
220 IF (MI .GT.0) GO TO 230
    IF (NFG .EQ. 0 .AND. O(1) .EQ. 0.) GO TO 230
    IF (NFG .GT. 0 .AND. J .LT. NR) GO TO 230
    KN=MH
    GO TO 250
230 KN=MH-MI
    IF (KN .EQ. 0) GO TO 260
    DO 240 IJ=1,KN
    XM(IJ)=XM(MI+IJ)
240 YM(IJ)=YM(MI+IJ)
    MI=MH
250 K=KN
    XM(K+1)=XA
    YM(K+1)=YA
    XM(K+2)=XB
    YM(K+2)=YB
    CALL LINE (XM,YM,K,1,0,0)
260 CONTINUE
270 CONTINUE
    IF (NB .EQ. 0) GO TO 340
    DO 330 I=1,NP
    MH=0
    N=1
    K=0
280 K=K+1
```

```
      IF(K .GT. NB) GO TO 330
      IF (NS(K) .NE. 1) GO TO 280
      KK=2**(NB-K)
  290 N=N+KK
      IF (R(N,NS(K)) .EQ. 0.) GO TO 330
      DO 320 J=1,2
      MH=MH+1
      IF (J .EQ. 2) GO TO 300
      MJ=N-KK
      MK=NS(K)-1
      GO TO 310
  300 MJ=N
      MK=NS(K)
  310 XM(MH)=Z(MG(MJ,MK),1)+XR
      YM(MH)=Z(MG(MJ,MK),2)+YR
  320 CONTINUE
      N=N+KK
      IF (N .LT. NR) GO TO 290
      IF (NFG .GT. 0 .AND. N .GT. NR) GO TO 330
      IF (NFG .EQ. 0 .AND. N .GT. (NR+1)) GO TO 330
      IF (N .GE. (NR+1)) N=1
      MH=MH+1
      MK=NS(K)-1
      XM(MH)=Z(MG(N,MK),1)+XR
      YM(MH)=Z(MG(N,MK),2)+YR
      K=MH
      XM(K+1)=XA
      YM(K+1)=YA
      XM(K+2)=XB
      YM(K+2)=YB
      CALL LINE (XM,YM,K,1,0,0)
  330 CONTINUE
C
C*************************************************************************C
C     CALCULATE AVERAGE DRAWDOWN FOR EACH ELEMENT AND PLOT IT ON THE GRAPH     C
```

```
C***********************************************************************************C
C
  340 LK=0
      MK=0
      XN=10.5
      YN=10.25
      XQ=11.
      ML=ME(II)
      IF(ME(II) .GT. 1) ML=ME(II)-1
      DO 600 KR=2,3
      DO 600 IR=KR,NP
      DO 590 IO=1,NR
      KJ=0
      IF(KR .EQ. 2 .AND. IR .GT. 2) GO TO 590
      IF (NEG .GT. 0 .AND. IO .EQ. NR) GO TO 360
      IF (NND(IO) .GE. (IR-1)) GO TO 360
      IF (IE .EQ. 0 .OR. NB .EQ. 0) GO TO 590
      KJ=1
  350 IF (KJ .GT. IE) GO TO 590
      IF(IR.EQ.NNP(KJ,4).AND.IO.EQ.NNR(KJ,4).AND.IS(KJ).EQ.3) GO TO 360
      KJ=KJ+1
      GO TO 350
  360 IF (NB .GT. 0) GO TO 370
      KLT=IO+1
      GO TO 420
  370 IF(ND(IO) .GE. IR .AND. R(IO,IR) .EQ. 0.) GO TO 590
      K=1
      IF (KR .EQ. 2) GO TO 410
  380 IF (IR-NS(K)) 410,400,390
  390 IF (K .GT. NB) K=NB
      IF (K .EQ. NB) GO TO 400
      K=K+1
      GO TO 380
  400 KK=2**(NB-K)
      KLT=IO+KK
```

```
      GO TO 420
  410 KK=2**(NB-K+1)
      KLT=IO+KK
  420 IF (KLT .GT. NR .AND. NFG .GT. 0) GO TO 590
      L=0
  430 IF (L .GT. 0 .AND. KJ .GT. 0) GO TO 590
      IF (KLT .GT. NR) KLT=1
      IF (KR .EQ. 2 .AND. IO .EQ. 1) GO TO 450
      IF (ME(II) .EQ. 1) GO TO 440
      LK=LK+NZ-1
      GO TO 470
  440 LK=LK+NZ-1
      GO TO 460
  450 LK=ML
      IF (ME(II) .GT. 1) GO TO 470
  460 NN=7
      IF (NOD(LK,7) .EQ. 0) NN=5
      SM=4.
      IF (NN .EQ. 5) SM=3.
      MM=1
      GO TO 480
  470 NN=8
      IF (NOD(LK,7) .EQ. 0) NN=6
      SM=4.
      IF (NN .EQ. 6) SM=3.
      MM=2
  480 SV=0.
      XM(1)=0.
      YM(1)=0.
      JV=0
      JVV=0
      DO 500 I=MM,NN,2
      SV=SV+(1./SM)*(S(NOD(LK,I),2))
      XM(1)=XM(1)+(1./SM)*(Z(NOD(LK,I),1))
      YM(1)=YM(1)+(1./SM)*(Z(NOD(LK,I),2))
```

```
      IF (I .EQ. MM) GO TO 490
      XP=XP-Z(NOD(LK,I),1)
      RM=RM-Z(NOD(LK,I),2)
      DL=(XP*XP+RM*RM)**(0.5)
      DL=5.*DL/RR
      IF (DL .GT. 0.7) GO TO 490
      JV=JV+1
      IF (DL .GT. 0.20) GO TO 490
      JVV=JVV+1
  490 XP=Z(NOD(LK,I),1)
      RM=Z(NOD(LK,I),2)
  500 CONTINUE
      XM(1)=(XM(1)+XR)/XB
      YM(1)=(YM(1)+YR)/YB
      IF (JV .GE. 2) GO TO 530
      CALL SYMBOL (XM,YM,0.07,3,0.,-1)
      IF (KR .GT. 2) GO TO 510
      XM(1)=XM(1)-(1./16.)
      GO TO 520
  510 XM(1)=XM(1)-(5./32.)
  520 YM(1)=YM(1)+0.05
      CALL NUMBER(XM,YM,0.07,SV,0.0,2)
      GO TO 580
  530 MK=MK+1
      IF (MK .LE. 20) GO TO 540
      XN=XN+2.5
      XQ=XQ+2.5
      YN=10.25
      MK=1
  540 IF (MK .GT. 1) GO TO 550
      IF (JVV .LT. 2) GO TO 560
      CALL SYMBOL (XM,YM,0.04,3,0.,-1)
      GO TO 570
  550 IF (JVV .GE. 2) GO TO 570
  560 YM(1)=YM(1)-0.02
```

```
      CALL SYMBOL (XM,YM,0.06,AN(MK),0.0,2)
  570 YN=YN-0.5
      CALL SYMBOL (XN,YN,0.14,AN(MK),0.0,2)
      CALL NUMBER (XQ,YN,0.14,SV,0.0,2)
  580 IF (NB .EQ. 0) GO TO 590
      IF (IR .NE. NS(K)) GO TO 590
      IF (R(KLT,IR-1) .EQ. 0. .OR. L .GT. 0) GO TO 590
      L=L+1
      GO TO 430
  590 CONTINUE
  600 CONTINUE
C
C***********************************************************************C
C     PLOT TITLE FOR THE GRAPH                                          C
C***********************************************************************C
C
      SV=TT
      IF (IV .EQ. 1) GO TO 610
      CALL SYMBOL(1.0,10.1,0.21,'AVERAGE DIMENSIONLESS DRAWDOWN FOR T=',
     *     0.0,37)
      GO TO 620
  610 CALL SYMBOL(0.9,10.1,0.21,'AVERAGE REAL DRAWDOWN FOR REAL TIME='
     *,0.0,36)
  620 XN=9.0
      CALL NUMBER (XN,10.1,0.21,SV,0.0,1)
C
C***********************************************************************C
C     REPOSITION THE PEN FOR SUBSEQUENT PLOTTING                        C
C***********************************************************************C
C
      CALL PLOT (20.,0.2,-3)
  630 CONTINUE
      RETURN
      END
```